Community and Thread Methods for Identifying Best Answers in Online Question Answering Communities

Thesis

COMMUNITY AND THREAD METHODS FOR IDENTIFYING BEST ANSWERS IN ONLINE QUESTION ANSWERING COMMUNITIES

GRÉGOIRE BUREL

MSc, University of Caen Normandy, 2008

BSc, Paris Descartes University, 2006

Thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Computer Sciences

Knowledge Media Institute

The Open University

2015

Grégoire Burel: *Community and Thread Methods for Identifying Best Answers in Online Question Answering Communities,* Thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Computer Sciences, © 2015

SUPERVISORS:

Harith Alani

Yulan He

Paul Mulholland

EXAM PANEL CHAIR:

Trevor Collins

EXAMINERS:

Stefan Rueger

John Breslin

Dedicated to my *mother*, my *wife* and my *daughter*.

ABSTRACT

Much research has recently investigated the measurement of quality answers in QUESTION ANSWERING (Q&A) communities in the form of automatic *best answer* identification. Previous approaches have focused on manual user annotations and diverse features based on intuition for identifying *best answers* and proved relatively successful despite considering *best answer* identification as a general classification problem.

Best answer modelling is generally distanced from community studies about what users regard as important for identifying quality content (i.e. user studies). In particular, previous research tends to only focus on the automatic aspects of *best answers* identification model by applying generic learning algorithms.

This thesis introduces the concepts of *qualitative* and *structural* design in order to investigate if features derived from community questionnaires can enrich the understanding of *best answer* identification in Q&A communities and if the thread-like structure of Q&A communities can be exploited for better results. Two different approaches for exploiting the thread structure of Q&A communities are proposed and two new, previously unstudied, features

are introduced. First, a measure of question complexity is introduced as a proxy measure of answerer knowledge. Second, different models of contribution effort are proposed for representing the answering reactivity of contributors.

The experiments are systematically conducted on datasets issued from three different communities that vary in size, content and structure. The results show that the newly proposed features allow for better understanding of what constitute *best answers*. The findings also reveal that the thread-wise algorithms and optimisation techniques created from the structural design methodology correlate with *best answers*. In general both structural and qualitative design appear to improve *best answer* identification meaning that structural and qualitative methods may improve unrelated classification tasks.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my supervisors, Dr Harith Alani, Dr Yulan He and Dr. Paul Mulholland for their patience and constant supervision throughout my research. Whenever I had an issue or a question they always had time for discussion and feedback. I would like to particularly thank Dr. Harith Alani for his though challenging questions that allowed me to focus on the fundamentals whenever I was diverting from my research. His ability to transform complex ideas into clear concepts showed me that anything can be tackled with the right formulation and coherent explanations. Thanks to Dr. Yulan He's invaluable technical help, I was able to frame and focus the most complex part of my thesis. Her critical help taught me how to design robust experiments and to discuss results critically. Finally, Dr. Paul Mulholland guidance showed me the importance of good writing in research. Hopefully, his ability to spot the smallest typos and grammatical errors made me a better writer.

I would also like to thank my friends and colleagues at the Knowledge Media Institute (KMi) and beyond who have supported me one way or another during the course of my studies.

Finally, I would like to thank my family and particularly my wife for supporting me over all these years. Their unconditional support kept me going throughout my studies. Last but not least, I would

like to thank my daughter for her supervising abilities and enthusiasm. Her smiles and cooing of encouragement cheered me up when I was finalising my dissertation.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF DEFINITIONS

ACRONYMS

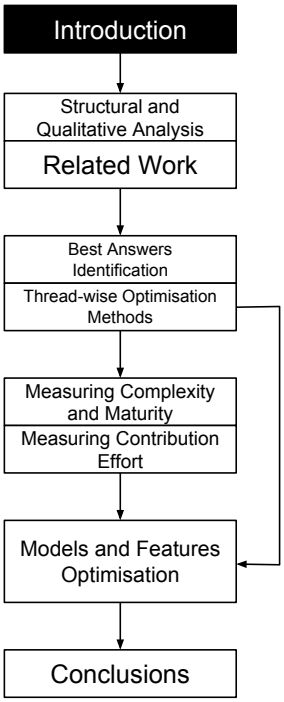
ADTree	Alternating Decision Tree
AES	Automatic Essay Scoring

ATM	AUTHOR TOPIC MODEL
AUC	AREA UNDER THE CURVE
B2B	BUSINESS-TO-BUSINESS
B2C	BUSINESS-TO-CONSUMERS
C2B	CONSUMERS-TO-BUSINESS
C2C	CONSUMERS-TO-CONSUMERS
CFS	CORRELATION FEATURE SELECTION
CHI	COMMUNITY HEALTH INDEX
CO	COOKING
CRM	CUSTOMERS RELATIONSHIP MANAGEMENT
DTM	DYNAMIC TOPIC MODEL
EC	ENQUIRY COMMUNITY
FPR	FALSE POSITIVE RATE
HITS	HYPERLINK-INDUCED TOPIC SEARCH
IGR	INFORMATION GAIN RATIO
IG	INFORMATION GAIN
IRT	ITEM RESPONSE THEORY
IR	INFORMATION RETRIEVAL
ITS	ISSUE TRACKING SYSTEM
JET	JOINT EFFORT TOPIC MODEL
JST	JOINT SENTIMENT TOPIC MODEL
KDE	KERNEL DENSITY ESTIMATION
LDA	LATENT DIRICHLET ALLOCATION
LTR	LEARNING-TO-RANK
ML	MACHINE LEARNING

MRR	MEAN RECIPROCAL RANK
NLP	NATURAL LANGUAGE PROCESSING
POS	PART OF SPEECH
Q&A	QUESTION ANSWERING
ROBUST	RISK AND OPPORTUNITY MANAGEMENT OF HUGE-SCALE BUSINESS COMMUNITY COOPERATION
ROC	RECEIVER OPERATING CHARACTERISTIC
SCN	SAP COMMUNITY NETWORK
SE	STACK EXCHANGE
SF	SERVER FAULT
SO	STACK OVERFLOW
SVM	SUPPORT VECTOR MACHINES
TPR	TRUE POSITIVE RATE
UGC	USER GENERATED CONTENT
αJET	AUTHOR JOINT EFFORT TOPIC MODEL
dJST	DYNAMIC JOINT SENTIMENT TOPIC MODEL

1 INTRODUCTION

In recent years, online QUESTION ANSWERING (Q&A) communities have seen a dramatic increase in popularity as a means to find answers to questions that cannot be solved directly by search engines. Nowadays, websites like STACK OVERFLOW (SO),[1] Quora[2] and Yahoo! Answers[3] attract together more than 52 million visitor a day.[4] With this increase in popularity, the management of the large amount of contributed content has become critical as community managers need to ensure that questions do not duplicate too often, that questions receive answers in a timely manner and that users can easily find questions to answer as well as answers to questions. Such issues have led to different areas of research including the matching of existing answers to new and existing questions and the task of recommending questions and answers to contributors (Chapter 3).



Although most of such websites support manual annotations for identifying *best answers*, which are effective solutions to particular questions, these ratings may be unavailable due to the labour intensive nature of such a process. For instance, in SO the proportion of questions with a best answer is only 56.8% out of 9.1m posted questions. The lack of *best answers* annotations for a large portion of answers in online communities means that the research

[1] *Stack Overflow,* `http://stackoverflow.com`.

[2] *Quora,* `http://quora.com`.

[3] *Yahoo! Answers,* `http://answers.yahoo.com`.

[4] *According to* `compete.com`, *Quora attracted more than* 7.9*m unique visitors in February 2015 while Yahoo! Answers had more than* 40.5*m visitors and* STACK OVERFLOW *(SO)* 3.8*m*

mentioned above tends to only apply to a small portion of the content posted on QUESTION ANSWERING (Q&A) websites unless automatic approaches are used for identifying best answers.

Most current approaches that have focused on the automatic identification of quality answers have been based on third-party annotations (i.e. out of community crowd sourced labels) rather than on using available *best answers* (i.e. answers that are identified as question solutions) annotations. Similarly, such works have widely ignored community studies concerning the factors that make quality answers, preferring to use less interpretable features such as n-grams. Existing approaches have also massively relied on standard classification models failing to take into account the thread-like structure of Q&A communities when designing classification models and feature normalisation methods.

Not taking into account the structure of such communities when identifying *best answers* may lead to less accurate results as answer identification is thread dependent and do not depend directly on all the answers contributed in a community .Similarly, by not taking into account user studies about answer quality, existing works generally rely on arbitrary features. The selection of better features could benefit from such type of studies.

In this thesis, new models for predicting *best answers* are proposed. Such models are motivated by user survey results that are used for designing new features that are expected to help the automatic identification of *best answers*. In particular two new complex features are designed as proxy measures of answering knowledge and user reactivity by the means of the concepts of user maturity and contribution effort. Besides these features, more standard predictors are

also used. Two different approaches are investigated for optimising these models using the thread-like structure of Q&A communities. This approach is based on the concept of thread features as well as the application of specific MACHINE LEARNING (ML) models instead of more conventional classification models.

Contrary to most previous works, the proposed models and features are evaluated on multiple datasets that vary in size, topics and structure. The evaluated communities include a small size *cooking* community, and two different *technical* communities.

The main contribution of the proposed work is the evaluation and design of features based on qualitative studies and the design of optimised identification models based on the structural analysis of Q&A communities. Such a methodology is defined in this thesis as *qualitative design* and *structural design*.

This chapter presents the different research questions, contributions and hypotheses investigated in this thesis as well as the structure of the thesis. Therefore, the main contributions of this chapter can be summarised as follows:

- The motivation for the area of investigation of this thesis is presented. In particular, the importance of the automatic identification of *best answers* is discussed.
- The research questions and hypotheses studied in this body of work are introduced. In particular, the idea of *structural and qualitative design* is discussed.
- The different contribution of the thesis are presented in the form of two different structural optimisation methods and two qualitatively derived features.

– The structure of the different chapters and sections is highlighted as well as the list of publications that emerged from the content of this thesis.

In the following sections, the motivation behind this thesis is highlighted, and research questions, hypotheses and contributions are discussed in more details.

1.1 MOTIVATION

With the growth of the web, users have been more keen to use community websites in order to fulfil their information needs rather than only relying on search engines. In this context, recent years have seen an increase in popularity of Q&A communities as they can be used for seeking answers to complex questions that cannot be answered directly by search engines.

Many different type of Q&A communities have been created for serving different purposes and targets. For instance, companies such as Apple,[5] SAP[6] and Microsoft[7] use Q&A platforms for both reaching out to their customers, providing low cost support and improving their brand perception.[8] Other general purpose Q&A communities have also been created such as Yahoo! Answers and Quora. More specific communities have also been designed to provide a forum for users to seek knowledge on particular topics. For example, the COOKING (CO)[9] communities is centred around cooking enthusiasts while the SERVER FAULT (SF)[10] website is about computer system administration.

[5] *Apple Support Communities*, http://discussions.apple.com.

[6] SAP COMMUNITY NETWORK (SCN), http://scn.sap.com.

[7] *Microsoft Community*, http://answers.microsoft.com.

[8] *Kietzmann et al. (2011)*; *Mangold and Faulds (2009)*

[9] COOKING (CO), http://cooking.stackexchange.com.

[10] SERVER FAULT (SF), http://serverfault.com.

The critical reliance of companies and users on Q&A websites means that such communities need to be monitored for making sure that such websites thrive and grow. Similarly, it is also important to provide methods that make these websites easier to search and use when a large amount of information is available.

1.1.1 *Sustainability of Question Answering Communities*

The sustainability of these communities is conditional to the ability of question askers and information seekers to find answers, and for answerers to find relevant questions. Consequently, such websites require tools for finding relevant questions and quality answers accurately. Fortunately, many existing platforms have integrated reputation systems[11] which allow the manual annotation by users of the solution or *best answers* to posted questions. However, due to the manual nature of such a task, *best answers* labels are not always available (for example users may forget to identify best answers or simply do not bother to acknowledge correct solutions), thus prompting the need for automatic methods for identifying such type of answers.

[11] Farmer and Glass (2010)

Besides the identification of *best answers*, other approaches have been proposed for helping communities to function properly such as question recommendation, expert identification and questions and answers retrieval (Chapter 3).

Nevertheless, all such approaches require the fundamental understanding and measurement of quality answers and *best answers* and therefore the need for identifying *best answers* automatically. For example, expert users are providers of quality answers, question

recommendation needs to recommend quality questions or focus on questions that have already been answered successfully, and, answer retrieval requires the measurement of answer quality and identification of *best answers*. All the above cases need some method for identifying quality content and by extension *best answers*.

Even though, there can be more than one correct answer for a given question, the idea of focusing on *best answer* is sensible as, for a question asker, it can be argued that a particular answer fulfil her personal needs compared to any other correct answer. In this context focusing Q&A answer analysis on the measurement of *best answers* is reasonable as on of the most important aim of Q&A communities is to ensure the satisfaction of question askers.

Overall, the previous observations reinforce the importance of *best answer* identification as a critical component for improving the sustainability of Q&A communities.

Research on the modelling of quality answers and *best answers* has attracted much research (Chapter 2 and 3). However, previous work has been split into qualitative user studies and the design of automatic models for identifying *best answers*. These two directions of studies produced disparate findings. In particular, existing automatic modelling work has generally ignored user studies results about answer quality factors and the possible integration of such findings in the design of their identification models. By identifying potential features from user studies, more accurate models may be constructed and a better understanding of the relations between user beliefs and data observations can be derived.

Existing work has also focused on the application of standard classification algorithms rather than the design of specific models that

account for the structure of Q&A communities. The major drawback of such an approach is the omission of question context when identifying *best answers* as within those models *best answers* are identified at the community level rather than at the question level. Therefore, the design of optimisations methods that take into account the thread-like structure of Q&A websites is important in order to obtain more accurate results.

1.1.2 *User Studies, Derived Features and Thread-wise Best Answer Identification*

As previously observed, current approaches for identifying *best answers* suffer from a lack of grounding for designing the features required for identifying best answers despite different qualitative studies in answer quality. Another issue is a lack of model optimisation taking into account community structure for identifying *best answers* that could be used for improving the automatic identification of *best answers*.

As a consequence, the area of research proposed in this thesis is the design and integration of *best answer* predictors based on qualitative studies into optimised *best answer* identification models. Such a methodology is characterised by the *qualitative design* of features and the *structural design* (i.e. structural optimisation) of identification models. Therefore, the proposed research can be split into three different areas: 1) User study of *best answer* predictors; 2) Modelling of features based on *best answer* predictors identified in users studies, and; 3) The design and optimisation of *best answer* identification model based on the structure of Q&A communities.

As a summary, the work presented in this thesis addresses the problem of automatically identifying *best answers* in Q&A communities by using structural and qualitative design. The novelty of the approach is based on: 1) The analysis of user beliefs about what makes a good answer and user, and how to characterise *best answers* and the design of associated features, and; 2) The usage and design of thread-based *best answer* identification optimisation techniques.

In the remaining sections of this chapter, the research questions studied in this thesis are presented as well as different hypothesis and the contributions to the state of the art. The thesis structure and methodology is also introduced.

1.2 Research Questions, Hypotheses and Contributions

The previous observations highlight the possible gains in designing automatic *best answers* identification models that take into account the structure of Q&A communities as well as the outputs of user studies. Such a qualitative and structural design approach is the key contribution of this thesis and the main area of investigation. Such a contribution is investigated as part of the main research question which is defined as follows:

Main Research Question 1. *Can* structural *and* qualitative design *improve the performance of automatic identification of* best answers *in online* Q&A *communities, and if so how?*

The evaluation and investigation of the above research question can be divided into two main areas: 1) The investigation of structural optimisation of *best answer* identification algorithms, and; 2) The investigation of features derived from qualitative studies about what community contributors associate with *best answers*.

Research Question 1.1. *Can structural optimisation techniques improve automatic best answer identification and if so how?*

Previous research has largely ignored the particular structure of Q&A communities and considered the automatic identification of *best answers* as a community-wide binary classification problem between *best answers* and *non best answers*. In this thesis, it is postulated that the structure of Q&A websites can be used for designing a better suited identification model as *best answers* are question dependent.

The main characteristic of *best answers* in Q&A communities is that it is considered that only one answer for a given question can be identified as a solution even though it is possible that more than one answer could help the resolution of a particular issue. Another property is the thread-like structure of such communities as the content is organised into questions with a number of associated answers. These observations lead to the following hypothesis:

Hypothesis 1.1. *Structural optimisations techniques that take into account the thread-like structure of* Q&A *communities can help the automatic identification of* best answers.

In order to test this hypothesis, two different structural optimisations techniques are explored based on the thread-like structure of the communities investigated in this thesis. First, the concept of feature normalisation is studied by designing *thread* features that represent the relative value of *best answer* predictors within the available answers of questions. The rationale behind this approach is that *best answers* are only evaluated against the answers posted to a particular question. Therefore, the relative value of predictors within an answering thread can be used efficiently for discriminating *best answers* from normal answers within a thread. Secondly, the application of LEARNING-TO-RANK (LTR) models is considered as only one answer needs to be identified as *best answer* for a given question. The idea is that ranking models can be applied to question threads sequentially in order to identify the most likely *best answer* for a question rather than the most likely *best answers* for all the questions posted in a community. Such optimisations are compared against a large set of *user* and *content* features and a few *thread* features on three different communities.

Accordingly, the contributions matching the research question on the impact of structural optimisation on the identification of *best answers* are:

- The introduction and evaluation of *user*, *content* and *thread* features for automatically identifying *best answers*.

- The introduction of a systematic structural approach for normalising features based on same features relations in answering threads (i.e. an approach for generalising *thread* features).
- The design and investigation of the applicability of three different thread based normalisation methods: proportional normalisation, order normalisation and normalised order normalisation.
- The evaluation of the performance of a pointwise LEARNING-TO-RANK (LTR) approach for automatically identifying *best answers*.
- The investigation of the impact of rank based features on *best answers* binary classifiers and pointwise LTR models on three different communities.

Research Question 1.2. *How do user beliefs about what makes quality answers compare to the other features that identify* best answers*?*

Existing research has been mostly divided into qualitative studies and quantitative models for identifying *best answers*. Such models have mostly ignored the potential advantage of using features based on community contributors' knowledge about their community. Such knowledge can help the identification of features that are associated with quality content and *best answers*.

Instead of only relying on intuition and previous research for deciding what features to use for designing models of *best answers*, this thesis proposes to partially rely on qualitative user studies and

questionnaires for identifying the features that may help *best answer* identification. The idea is that community contributors are the most suitable estimators for indicating what makes their community worthy, their contributors and answers useful. Asking the contributors directly can help determining the set of features for designing and evaluating models for automatic identification of *best answers*. This observation can be summarised as the following hypothesis:

Hypothesis 1.2. *Community contributors' belief about what makes quality answers can be used for identifying and designing features that correlate with* best answers.

For addressing the above hypothesis, existing user studies are reviewed and a user questionnaire is designed in order to obtain a better idea of how users perceive the value of answers in Q&A websites. Based on the questionnaire results two features are identified, designed and integrated into a *best answer* model in order to understand how such features perform for identifying *best answers*. The two features extracted from the user studies are *question complexity* and the derived metric of user *maturity* as well as *contribution effort*; a measure that estimates the amount of work required for users to contribute. Such measures are proposed as the respective proxy measures to the users' ability to learn new things and be knowledgeable, and the ability of users to reply promptly. The ability of each measure to be used as a proxy measure of knowledgeable users and user reactivity is investigated in separate research questions.

The contributions from investigating the research question on the identification of contributors' beliefs concerning what makes best answers are summarised as follows:

- The review of existing qualitative studies and survey about the perceived value factors of Q&A communities and best answer.
- A questionnaire about the perceived factors for best answer identification in different Q&A communities.
- The identification and design of question complexity and user maturity as a potential important factor for identifying best answers.
- The identification and design of contribution effort as a potential important factor for identifying best answers.
- The investigation of the impact of qualitative features on automatic best answers identification models on three different communities.

Although, the qualitative features are identified from the user studies presented in the following chapter, each of such metrics spawn their own research questions as they seek to model the two particular features derived from the qualitative design approach investigated by the previous research question. The first feature is *question complexity* and the associated measure of *maturity*; a measure of answering ability and user knowledge. The second feature is contribution effort a measure that models the amount of work required by individual users to contribute that can be used for estimating the reactivity of a particular answerer. The design of such features leads

to the two following research questions.

Research Question 1.3. *Can question complexity and user maturity be used for measuring the ability of users to learn new things and being knowledgeable and if so how ?*

Based on the results of the user studies, the concept of question complexity and community maturity is proposed as a method for measuring the ability of users to learn new things and a proxy measure of knowledge. Rather than simply modelling the knowledge of users through the simple usage of reputation systems (Chapter 2), this thesis proposes to use question difficulty and the increasing number of difficult questions that have been answered, to model user knowledge as they seek to learn new things (i.e. user maturity). The idea is that as users *mature*, they become more knowledgeable and are able to answer more difficult or *complex* questions. This idea is summarised by the following hypothesis:

Hypothesis 1.3. *Knowledgeable users are more likely to answer or ask complex questions.*

In order to validate the previous hypothesis, complex questions must be differentiated from easy questions. For identifying complex questions, questions are manually annotated and different complexity models are constructed using *askers*, *answerers*, *questions* and *answers* features. Then, a measure of maturity based on question complexity is proposed and the resulting models are then evaluated against user reputation and user community involvement in

order to validate the relation between user knowledge and maturity. Besides such contribution, a community agnostic complexity measure called *Omega* is proposed.

The contributions relating to the design of question complexity and user maturity measures are summarised as follows:

- A definition of question complexity is introduced and the relation between question complexity, community involvements and reputation is studied.
- The influence of *asker*, *answerer*, *question* and *answer* features on question complexity prediction is studied.
- The concept of user maturity, a proxy measure of user knowledge is introduced and evaluated.
- A community agnostic complexity metric that can be used on different datasets is introduced.

Research Question 1.4. *Can contribution effort be used for modelling the reactivity of community users in contributing particular answers and if so how?*

Besides the previous feature, the importance of community reactivity was highlighted in community studies as a potential good indicator of best answers (Chapter 2). Although, time-to-response information can be used for estimating the reactivity of answerers, such metrics do not account for the hidden amount of time a user required for contributing to a particular answer. In this context, this thesis proposes to estimate the relative amount of time used for contributing to a particular post by attributing effort values to

the individual words that form a particular contribution using available time-to-answer information. Based on such information, the amount of work or effort associated with individual words can be used for estimating the hidden cost of user contributions (Chapter 7). This idea is summarised in the following hypothesis:

Hypothesis 1.4. *User reactivity can be estimated from the amount of effort required for generating the words that form an answer.*

The validation of the previous hypothesis requires the design of different models that estimate the amount of effort associated with the vocabulary of individual users and then study the relation between time-to-answer information and contribution effort. In this thesis, different models of effort are proposed based on two distinct ideas: 1) The concept of Stanine,[12] a grading measure used in examination marking schemes based on z-scores that can be used to normalise the amount of effort associated with individual contributions, and; 2) Topic models, a family of Bayesian models that can be used for learning latent topics and other features from textual content.

[12] *Thorndike (1982)*

The contributions relating to the design of contribution effort and community reactivity are summarised as follows:

- The concept of contribution effort, a value representing the level of labour and time required for contributing or posting to a community is introduced.
- Two measures of effort based on the concept of Stanines (STAN and ASTAN) are introduced.

– The Joint Effort Topic (JET) model and its authored version, the Author Joint Effort Topic (αJET) model designed for balancing out STAN and ASTAN effort modelling weaknesses are proposed for modelling contribution effort.

– The evolution of community effort in two different communities is investigated and the relation between effort and community reactivity is studied.

1.3 Thesis Methodology and Structure

The methodology used in this thesis for evaluating the proposed hypotheses and research questions relies on a three to four step process that is followed for each of the proposed experiments (Figure 1). First, an extraction phase is conducted for acquiring the different features required for generating the models presented in the corresponding chapters. For example, in the fourth chapter, *user*, *content* and *thread* features are extracted. In the fifth chapter, a similar extraction process is combined with a feature normalisation approach. Secondly, a modelling phase is performed where a model is generated based on the features extracted in the first phase. For example, in the fourth chapter, a supervised binary classifier is trained for learning a *best answer* classifier. Thirdly, an evaluation phase to evaluate and analyse the ability of the model to perform accurately. For example, in the fourth chapter, the ability to identify *best answers* is evaluated. Finally, for the chapters that deal with the research questions discussed in section 1.2, a fourth phase is added for investigating if the developed models validate the hypotheses associated with the research questions. For example, in the fifth

chapter, the usefulness of structural design is evaluated by comparing best identification models that include structural optimisation compared to non optimised models.

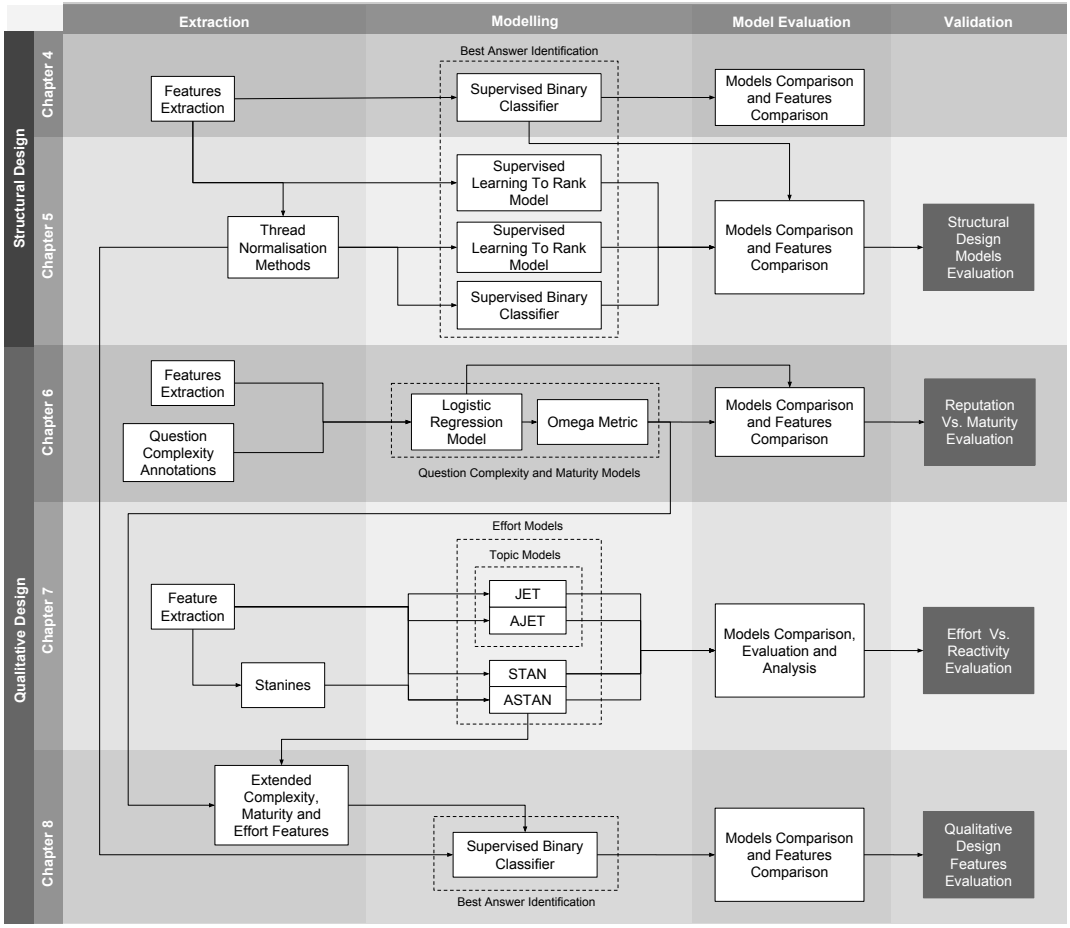


Figure 1: Organisation and relations between the chapters and the research conducted in this thesis.

This thesis is divided into several parts where the results from different chapters are reused as part of further experiments while following the three to four step process described above. The relation between each experiment and research question is highlighted in Figure 1.

The content of this thesis is divided into four different parts and nine chapters. In the first part of this body of work, the motivation for using structural design and the features retained by the qualitative design process are discussed as well as existing work. In the

second part, the research question about structural design is evaluated while the third part evaluates the impact of qualitative design. Finally the last part discusses the results of this thesis and future work. A more detailed discussion of the content of each of these parts is described in the following sections.

1.3.1 *Part I — Background and Exploratory Survey*

In Chapter 2, the different types of online Q&A communities are discussed as well as their structure in order to better understand and motivate the proposed integration of structural design into *best answer* identification models. Chapter 2 also defines key concepts and study what users consider important in ENQUIRY COMMUNITY (EC) by reporting the results of a questionnaire that was sent to different communities. The results of such a questionnaire are then used for the selection of features to be selected as part of the qualitative design methodology followed in the proposed research.

Following that chapter, Chapter 3 reviews existing research on content quality, user expertise and contribution effort as well as existing research on *best answer* identification.

1.3.2 *Part II — Structural Design Optimisation*

In Chapter 4 and Chapter 5, the impact of structural optimisation on automatic *best answer* identification is investigated. In Chapter 4, a standard binary classification model for identifying *best answers* is designed and evaluated. This chapter is also used for introducing

the concept of *thread* features. In Chapter 5, two distinct structural optimisations approaches are presented and evaluated in order to determine if structural design improves the automatic identification of *best answers* compared to the model presented in Chapter 4.

1.3.3 Part III — Qualitative Design Optimisation

Chapters 6 to 8 present and evaluate two different features based on the qualitative design methodology introduced in this thesis and investigate if qualitative design improves the automatic identification of *best answers*.

In Chapter 6, how users mature over time is proposed as a measure of the ability of users to learn new things, a proxy measure of user knowledge and reputation. The proposed method for measuring the maturity of users is introduced by defining different models that represent complexity of questions. This chapter also investigates if user maturity can be effectively used as a proxy measure of user reputation.

In Chapter 7, different models are presented for identifying the amount of work required by individuals to contribute to particular questions based on contribution patterns. Such a measure is proposed as a proxy measure of community reactivity. In this chapter, the effort of users is derived based on the evolution of user vocabulary over time. The resulting models are evaluated against their ability to identify reactive users.

Finally, in Chapter 8 the optimised models presented in Chapter 5 are revisited by integrating the newly introduced complexity, maturity and effort measures. The impact of each of these features on *best answer* identification is also analysed. In particular, this chapter investigates if the features that were derived from qualitative analysis correlate with *best answers*.

1.3.4 *Part IV — Conclusions and Future Work*

In the last part of this thesis (Chapter 9), conclusions are drawn by summarising-up the findings, limitations and potential future work. In particular the limitation and advantages of the qualitative and structural design proposed in this thesis are discussed and some research direction towards the automatic recommendation of question to potential answerers discussed.

1.4 PUBLICATIONS

The chapters of this thesis are based on different publications. These publications are listed below:

CHAPTER 2: Matthew Rowe, Harith Alani, Sofia Angeletou, and Grégoire Burel. *Report on Social, Technical and Corporate Needs in Online Communities*. Technical Report 3.1, ROBUST, 2011.

CHAPTERS 4 AND 5: Grégoire Burel, Yulan He and Harith Alani (2012). *Automatic identification of best answers in online enquiry communities*. In: 9th Extended Semantic Web Conference (ESWC '12), 27-31 May 2012, Crete, Greece.

CHAPTERS 5 AND 9: Grégoire Burel, Yulan He, Paul Mulholland and Harith Alani (2015). *Modelling Question Selection Behaviour in Online Communities*. Poster In: Web Science Companion Proceedings of the 2015 International Conference on the World Wide Web (WWW '15), 18-22 May 2015, Florence, Italy.

Grégoire Burel, Paul Mulholland, Yulan He and Harith Alani (2015). *Predicting Answering Behaviour in Online Question Answering Communities*. In: 26th Conference on Hypertext and Social Media (HT '15), 1-4 September 2015, Cyprus.

CHAPTERS 6 AND 8: Grégoire Burel and Yulan He. 2013. *A question of complexity: measuring the maturity of online enquiry communities*. In: 24th ACM Conference on Hypertext and Social Media (HT '13), 1-3 May 2013, Paris, France.

CHAPTERS 7 AND 8: Grégoire Burel and Yulan He. 2014. *Quantising Contribution Effort in Online Communities*. Poster In: Web Science Companion Proceedings of the 2014 International Conference on the World Wide Web (WWW '14), 7-11 April 2014, Seoul, Korea.

Part I

BACKGROUND AND EXPLORATORY SURVEY

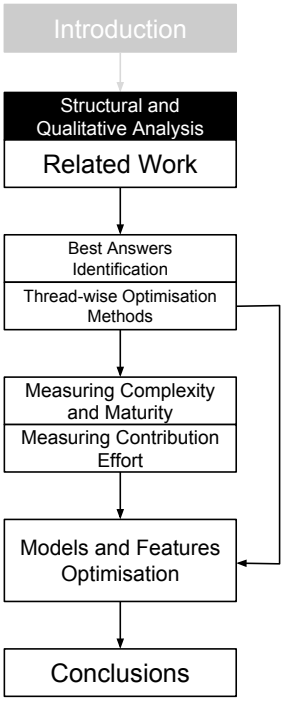
2

STRUCTURAL AND QUALITATIVE ANALYSIS

Previous research on the identification of *best answers* has focused on either the qualitative analysis of such communities or the creation of identification models that do not take into account their particular structure even though such information could be used for providing more accurate models.

This thesis proposes to study the integration of user studies about user beliefs on the constituents of *best answers* and the integration of the structure of Q&A communities for providing better models of *best answers* (RQ1).

In order to narrow down the area of investigation of the proposed approach, this chapter discusses the specific structure of Q&A communities for identifying elements that can be used for improving *best answer* identification models. This chapter also presents a user study of *best answer* indicators resulting in the selection of three different predictors that are investigated in the rest of this thesis.



Following the user study and structure analysis of Q&A websites, the focus of the research is concentrated on *thread-wise optimisation* and on the design of a *question complexity* and *user maturity* metric as well as on a model of *contribution effort* as they appear to be highly related to high quality content.

This chapter is divided into seven sections. First, the main contributions and rationale of this chapter are discussed. Second, the qualitative and structural design methodology used in this thesis is presented. Third, the designs and structures of online communities and Q&A communities are discussed and their specificities highlighted. In the fourth section, results of a qualitative user study about community value are reported and discussed against previous surveys. In the fifth section, common experimental approaches and evaluations methods are discussed. Finally, the datasets analysed in this thesis are presented and the chapter summarised.

2.1 INTRODUCTION

As observed in the previous chapter, the main idea of this thesis is to integrate user beliefs for designing *best answers* predictors and structural information for creating more accurate models for identifying *best answers* in Q&A communities (RQ1).

In order to identify what types of structural optimisation can be performed, the structure of online communities and Q&A communities needs to be investigated. Similarly, for identifying what types of features to design, this chapter also presents a qualitative study

on two online communities for better comprehending what makes a community valuable and quality answers.

This chapter also gives some background concerning commonly used methods for conducting experiments and evaluation methods for helping the understanding of the related work discussed in the following chapter and the experiments conducted in the rest of this thesis.

As part of this thesis, more than one community is analysed in order to better understand how quality factors hold between communities. This study is different to most previous work that have generally focused on single communities. Since the focus of the work is on more than one community, this chapter is also used to introduce each of the datasets employed in our proposed research.

Based on the structural analysis of Q&A communities and different qualitative studies of answer quality, distinct approaches can be proposed for both creating structural model optimisations and features created from qualitative observations. In this chapter two structural optimisation areas of investigation are highlighted and two new features are proposed based on user studies results. First, the thread-like structure of Q&A communities suggests the usage of algorithms that take into account these particularities such as thread-wise feature normalisation methods and specifically designed classification models. Secondly, the qualitative observations suggest the design of features that represent answerers' knowledge and ability to contribute promptly.

In summary, the contributions of this chapter are:

– The presentation of the structural and qualitative design methodology employed in this thesis.

- A discussion about the different categories of online communities and the types of Q&A platforms.
- A description of the structure of Q&A communities and the definition of the key concepts such as *question threads* and *best answers*.
- A qualitative user study on two online communities about the factors that make *best answers*.
- The selection of two different structural optimisation area of investigation and the introduction of two different qualitative features.
- A description of commonly used experimental approaches and evaluation methods.
- The presentation of the communities and datasets used in this thesis compared to previous works.

2.2 Improving Best Answer Identification using Structural and Qualitative Design

The idea of using the structure of data for improving the accuracy of *best answer* identification is motivated by the fact that Q&A communities are highly structured websites where content is organised according to a particular hierarchy and that each question has at most one *best answer* (Section 2.3.3).

In this context, it becomes possible to create custom methods that include such a particularity instead of only relying on standard classifiers that do not account for them.

The identification of *best answers* may also be improved by the usage of features that correlate with *best answers*. Unfortunately, designing such features is mostly based on the intuition that a particular feature may be useful for such particular tasks. Moreover, the number of features that can be designed is potentially limitless. As a consequence, a systematic method for guiding the development of such predictors may be beneficiary to the aforementioned approach of feature design.

The method proposed in this thesis is to analyse the structure of Q&A communities in order to identify what particularities can be used for creating optimised models for automatically identifying *best answers* and to survey the contributors of online communities about what they consider to be important factors of quality answers and *best answers*. Based on such user surveys, it is then possible to design features that are grounded in users beliefs rather than in intuition. Such an approach can potentially provide more accurate features compared to intuition-based methods.

The two mentioned methods form the *structural and qualitative design* methodology approach introduced in this thesis. First, a study of the structure of Q&A communities is used for determining what type of structural optimisation methods can be used. Second, user studies are performed for identifying the factors that correlate with *best answers*. Then, based on these observations, a set of features can be designed that model such *best answers* factors. Each of these methods correspond respectively to the *structural design optimisation and qualitative design approach* presented in this body of work.

In the following sections, the structure of Q&A communities is investigated and two different optimisation approaches selected. Then, a user study about what makes *best answers* is discussed and three different factors are retained. Using the selected factors, two different features are proposed. Such features are then compared with existing approaches (Chapter 3) before being introduced in more detail in the following chapters (Chapter 6 and 7)

2.3 Design and Structure of Q&A Communities

A wide variety of communities exist with different aims and structures. In this thesis the focus is on Q&A websites where users seek answers to different issues. One focus of the proposed research is to analyse if the integration of observations about the structure of Q&A communities can be used for improving the identification of *best answers*.

Previous research has mostly used algorithms and methods that do not necessarily take into account the structure and particularities of Q&A communities by relying mostly on non-specific features normalisation methods and standard classification models (Chapter 3).

In order to better understand what type of structural optimisation can be used for improving *best answer* identification, it is necessary to present the specificities of Q&A websites and how they differ from other type of online communities. Moreover, besides characterising Q&A communities, it is also important to define

fundamental concepts more formally as they are used extensively through this thesis such as the concept of *best answers* and *question/answering thread.*

Before defining what are Q&A communities, it is important to present the concept of online communities as the communities investigated in this thesis are a specific type of online community. The concept of online community is closely related to the notion of virtual community defined by Porter[13] since online communities are virtual communities where user interactions occur on the Internet. These two notions may be defined as follow:

[13] Porter (2004)

Definition 2.1 (Virtual Community)**.** *"A virtual community is defined as an aggregation of individuals or business partners who interact around a shared interest, where the interaction is at least partially supported and/or mediated by technology and guided by some protocols or norms".*[14]

[14] Porter (2004)

Definition 2.2 (Online Community)**.** *An online community is a virtual community where user interactions are mediated via the Internet.*

Consequently, online communities can be seen as the transposition of real world communities (i.e. communities that exists outside virtual environments) to an online setting where people come together to contribute content about particular interests or other similarities.

2.3.1 *Types of Online Communities*

Different approaches have been proposed for classifying online communities.[15] Some methods use the topology of online communities,[16] whereas; others have proposed to categorise them according to their media genre.[17]

[15] *Porter (2004); Bishop (2009); Kaplan and Haenlein (2010)*

[16] *Porter (2004)*

[17] *Bishop (2009)*

For simplicity this thesis uses the classification method proposed by Kaplan and Haenlein[18]. According to them, virtual communities can be divided into six different classes: 1) Blogs; 2) Social networking sites; 3) Virtual worlds; 4) Collaborative projects; 5) Content communities, and; 6) Virtual game worlds. Since virtual communities and online communities are highly related, such classification largely applies to online communities. Therefore, a similar classification can be used by considering the following categories:

[18] *Kaplan and Haenlein (2010)*

Blogs: Blogs are websites that present posted content in reverse chronological order. They are the "Social Media equivalent of personal web pages".[19] Typically, blogs have mostly a one way communication channel where users can reach their audience by publishing posts and obtaining feedback by allowing comments. Most of these websites are used for personal expression and by medias or companies for communicating informal newsworthy information. Many different bloging websites exists such as Tumblr[20] and blogger.[21]

[19] *Kaplan and Haenlein (2010)*

[20] *Tumblr, https://www.tumblr.com.*

[21] *Blogger, https://www.blogger.com.*

Content Communities: Content communities are websites that are centred around the concept of media sharing such as videos and pictures. These websites usually do not support complex interaction types and may only offer some method for users to comment on content.

Social Networking Sites: Social networking sites are perhaps the most active type of communities. A social network is mostly designed for users to communicate with each other and to discuss topics that tend to be personal and private. In particular, such websites usually encourage the creation of personal profiles and restrict the communication between members by allowing users to decide who can see what they contribute and who can communicate with them.

Virtual Worlds: Virtual worlds are websites that provide experiences that mimic real life interaction. Usually set in 3D environments, these communities are used by users or companies to communicate as if they where in the real world using 3D avatars that represent themselves.

Collaborative Community Projects: Collaborative communities include a wide range of websites where users come together and generate content towards a common aim or in order to help each other. The main element of such websites is their collaborative aspect. Such an aspect enables complex content to be created that cannot be effectively produced without the existence of a community.

Table 2: Types of online communities.

Type	Examples
Blogs	Tumblr, Blogger, Wordpress.
Social Networking Sites	Facebook, Twitter, Google+, MySpace, Linkedin.
Virtual Worlds	Second Life, Habbo.
Collaborative Communities	Wikipedia, Yahoo Answers, Stack Exchange, Ask.
Content Communities	YouTube, Flickr.
Virtual Game Worlds	World of Warcraft, Eve Online.

Virtual Game Worlds: Virtual game worlds are communities focused around games or created from multi-player games interactions. They usually share some similarities with virtual worlds except that players aims and avatars are predetermined by a set of rules. Moreover, contrary to virtual worlds, users are usually encouraged to play a particular role that may differ from their own personality.

Social networking sites are communities where users are mostly interested in inter-user communication whereas collaborative platforms let users contribute content around particular topics of interest. A few examples of such communities are given in Table 2. This thesis is concerned with a particular type of collaborative communities where users seek knowledge on particular topics by answering each other questions. The following section discuss the different incarnations of such Q&A communities as well as their structure.

2.3.2 *Types of Q&A Communities*

Q&A communities are a particular type of collaborative project community where user collaborate in order to answer each other questions. In this thesis such communities are defined as the following:

Definition 2.3 (Question Answering Community)**.** *Question Answering (Q&A) websites are a type of collaborative project community composed of askers and answerers looking for solutions to*

particular problems. In these communities, askers seek answers to their questions whereas answerers reply to the questions asked by askers. Askers and answerers are not mutually exclusive groups and users may be askers and answerers at the same time.

Many different types of Q&A communities have been set up for targeting different needs and types of users. For instance, general Q&A websites such as Yahoo Answers[22] target topics of general interest and allow free communication between users whereas specific systems like help desks are designed for restricting who can ask and answer questions. Nevertheless, all existing Q&A communities act as support communities where users look for help or advice on particular issues.

[22] *Yahoo Answers*, http://uk.answers.yahoo.com

Due to the versatility of Q&A systems, businesses have started to use Q&A systems for communicating with their user base and customers in an attempt to leverage cheap or free labour, improve their brand perception and increase their communication strategy.[23]

[23] *Kietzmann et al. (2011); Mangold and Faulds (2009)*

With the diversification of the usage of Q&A systems, different approaches have been taken for hosting and representing such communities depending of the type of communities targeted as well as the parties involved in question and answering activities.

For example, companies have created support communities where users can either communicate with each other or with company employees. In this particular domain, four different forms of communication are generally preferred depending of the parties involved. In this thesis the following types of asker/answerer relations are

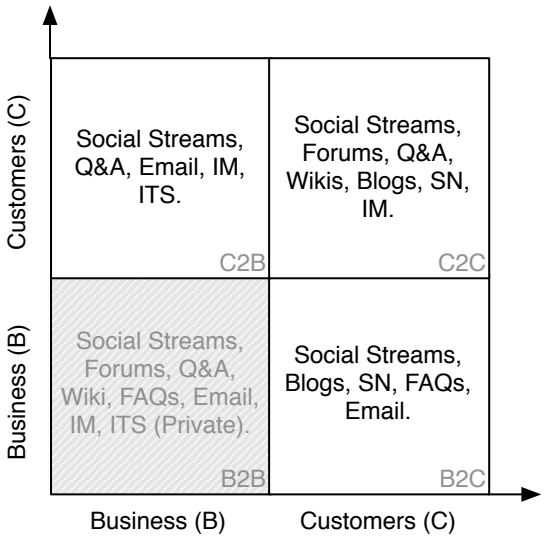


Figure 2: Enquiry Channels in Online Business and Customer Communities.

identified (Figure 2): 1) BUSINESS-TO-BUSINESS (B2B); 2) BUSINESS-TO-CONSUMERS (B2C); 3) CONSUMERS-TO-BUSINESS (C2B), and; 4) CONSUMERS-TO-CONSUMERS (C2C).

B2B channels are particular communications that occur within a company or between companies. For example, it can be a company social network where employees can communicate with each other. B2C communication happens when a business directly contact a customer. Rather than asking questions to customers, businesses may answer customer questions that are not directly directed to them. For example, a company might announce a new product functionality or address common users questions. C2B is the traditional vector of communication for customer support. In this context, product users contact a business for obtaining answers and businesses are the only actors able to answer enquiries. Finally, C2C relations involve communications where customers answer other customers questions. Very often, C2C support also involves some degree of C2B relations where a mix of customers and company employees answer customer questions. For instance, the SAP

COMMUNITY NETWORK (SCN) forums[24] are good examples of such a type of relationships.

[24] SCN, http://scn.sap.com.

Integrated Communities and Third Party Communities: Although, the previous discussion was focused on businesses organising their own community support, many are actually managed and created by the people requiring such support. For example, the SO[25] Q&A website was created in 2008 for helping users that have particular software programming issues. Online customers and business communities can be defined as groups of users sharing interests and knowledge about particular businesses, products and services. In this context, it is required to distinguish integrated communities from third party communities. On the one hand, integrated communities are groups created and fully or partially managed by businesses. On the other hand, third-party communities exist outside the direct control of businesses. For instance, the SCN community is an integrated business community whereas the STACK EXCHANGE (SE) community is a third-party community. Such a difference highlights the different relations existing between communities and companies. Within integrated communities, companies are able to moderate content. Integrated communities are also part of the public image of a particular business meaning that any community action is directly associated to company activities.

[25] *Stack Overflow*. http://stackoverflow.com.

Help Desks: Although many different types of platforms can be used for managing business communities, few systems are designed for managing the particular needs of Q&A and support communities. Companies traditionally manage customer issues using ISSUE

TRACKING SYSTEMS(ITSS). An ITS is a system designed for collecting and managing issues until they are solved or declared irrelevant. In such a setting, a technician, typically a business employee collects details from customers then decides if the problem is valid or not. Then, different tasks are performed for solving the issue until a solution is found. When a solution is reached, the issue is fixed by communicating the solution to the user. ITSs are usually bundled in Help Desk software, systems that are designed for collecting customer issues and solutions. Although a typical ITS workflow cannot generally be used as ECs software directly in the particular case of completely outsourced support communities, it can be used for dealing with customers to employee support where employees answers questions from users.

Discussion Groups and Forums: Nowadays, discussion groups and forums are probably the most used systems for managing C2C communities. Discussion groups encompass a large set of online discussion software where users can post information to each other and discuss particular topics collaboratively. Some examples of discussion software include mailing lists, forums and newsgroups. Forums are very popular discussion groups platforms due to their long history (earliest online forums date back to 1994 and are therefore very popular online support platforms.). Forums organise discussions in threads, a collection of posts organised in topics and in chronological order. For historical reasons, forums are widely used for managing support communities. For example, the SCN forums represent a community of users discussing issues related to SAP products. Although SCN users are engaged in peer to peer discussions, the SCN community is organised by SAP. Other support

communities include the third-party community Cable Forum[26], the integrated Xerox Customer Support Community forums[27] and the Apple Support Communities[28]. Most of the existing integrated communities use particular forum software that has extended capabilities designed for the specificities of CUSTOMERS RELATIONSHIP MANAGEMENT (CRM) such as Jive Engage[29] and Lithium[30]. Typically, customer oriented forums contain features found in community Q&A websites. They rely on *reputation* models designed for identifying valuable answers and influential users.

[26] *Cable Forum*, http://www.cableforum.co.uk/

[27] *Xerox Customer Support Community forums, the* http://forum.support.xerox.com.

[28] *Apple Support Communities*, https://discussions.apple.com

[29] *Jive Engage*, http://www.jivesoftware.com.

[30] *Lithium*, http://www.lithium.com.

Q&A Platforms: The focus of this thesis is mainly on Q&A platforms such as Yahoo Answers, Quora[31] and Stack Exchange. These communities have a very different structure compared to communities relying on forums for performing their community support. Contrary to bulletin board systems, Q&A websites are designed for avoiding conversational behaviours (i.e. questions that generate opinions rather than factual answers) as they try to make sure that answers can be identified easily. A typical Q&A system differentiates between two types of content: *questions* and *answers*. Each question thread contains a unique question and multiple self-contained answers whereas a forum thread can contain multiple questions and answers spanning multiple posts. In addition, many of these systems tend to include manual *reputation* mechanisms for distinguishing good answers from lower quality posts. In this regard, Q&A platforms are generally efficient for supporting Q&A communities. For example, the SCN community is supported by forums and the SE users employ a Q&A platform. Although the SCN forums have extended the core features of traditional bulletin board software for

[31] *Quora*, http://www.quora.com.

Table 3: Relative comparison of the characteristics of different social platforms.

Characteristic		Forums		
Type	Name	Jive Forums (SCN)	Jive Engage	SE
User	Profile	•	•	•
	Bookmarks	–	•	•
	Accepts	○	○	•
	Points (Reputation)	○	○	•
	Classes (Mod/Admin)	○	○	○
	Levels (Badge Types)	○	○	○
	Achievements (Badges)	–	•	•
	Abilities (Topic of Interest)	○	○	•
	Rewards (Bounties)	○	○	•
	Leading Board	○	•	•
Thread	Views	•	•	•
	Votes	○	•	•
	Lock (Not Editable)	○	○	•
	Sticky (Importance)	•	•	–
	Bookmarks	–	•	•
	Categories	•	•	–
	Tags	–	•	•
Question	Status (Open/Closed)	•	•	•
	Votes	–	○	•
	Comments	○	○	•
	Modification	○	○	•
Answer	Accepted	•	•	•
	Votes	○	•	•
	Comments	○	○	•
	Modification	○	○	•

Abbreviations: • = Yes. ○ = Limited/Partial. – = No.

including their own *reputation* framework, the SCN supports much less features compared to the SE network (Table 3).

In general, *reputation* features help the identification of quality content and *best answers*. Nevertheless, *reputation* features are only useful when they are accessible to all the produced content. Because *reputation* systems are mostly manual, *reputation* information are generally not attached to all community posts and users. This observation motivates the need of providing methods for automatically identifying quality answer and *best answers*.

2.3.3 *Structure of Q&A Platforms*

In this thesis, the focus is on three different communities (Section 2.6). One of the studied communities, the SCN forums use discussion forums as a backend whereas the other communities (SERVER FAULT (SF) and *Cooking*) both rely on the same custom Q&A platform that is specifically designed for asking and answering questions.

Although both systems share some differences in terms of structure and design, they both have the concept of *question/answering thread* and *best answer*. Such a general structure can be generally found in all existing platforms and it can be argued that the observations made in this section can be transposed to other communities.

The existence of *question/answering thread* derives from the fact that Q&As community allows for more than one answer for each question. In this context, it is useful to distinguish single answers from the set of the answers that are related to a particular question. In this thesis, such set of answers is refereed as *thread*:

Definition 2.4 (Question/Answering Thread)**.** *In Question Answering (Q&A) websites, a thread is the set of answers associated with a given question. For instance a question that contains five answers has an answering thread of length five. Depending on the type community, threads may have different structure (e.g. nested or flat hierarchies).*

In Q&A communities, each answer contributes to a particular question and may not solve a particular issue. As a result, Q&A communities allow for the identification of the unique correct answer to a question even when multiple answers are accurate within an answering thread. More precisely, the correct answer is the answer that solve a user issue and is the most useful answer for the question asker. A community can provide methods for rating answers or allow questions askers to identifying the best solution or *best answer* to their problem:

Definition 2.5 (Best Answer)**.** *In Question Answering (*Q&A*) websites, a best answer is an answer that is identified as the correct solution to a particular question. In general, each thread has at most one* best answer *that is labelled as such by the contributor that asked the question.*

The previously defined structure of Q&A communities means that each question has at most one *best answer* and that such an answer only exists within the answers that are found in the corresponding answering thread to a question. Therefore, a logical approach for improving the automatic identification of *best answers* is to take into account: 1) The fact that only one answer to a question may be identified as *best answer*, and; 2) The fact that the answers to particular questions are grouped together.

Another important concept related to *best answer* is the idea of *quality answer* as for a given question, it is possible that more than one answer may provide some help toward a question solution. In

this context a quality answer may be defined as follows:

Definition 2.6 (Quality Answer)**.** *In Question Answering (Q&A) websites, a quality answer is an answer that is helpful, sound and participates towards the resolution of a given question. It is possible for a quality answer to be also a best answer.*

As a summary, Q&A communities are centred around the concept of answering threads where questions have a set of answers and where a unique answer can be identified as a *best answer*. This thread-like structure is particularly important as such organisation can be potentially used for improving *best answer* identification by filtering out answers that are not associated with a particular question and only focusing on the comparison between answers of a same thread.

2.3.4 Reputation and Answer Metadata

The identification of *best answers* within existing platforms mostly rely on some form of a *reputation system* which allows users to rate or vote for the content and users that they consider valuable. The concept of *reputation* can be defined as follows:

Definition 2.7 (Reputation)**.** *Reputation is a measure used to make a value judgement about an object or person. For example, such value judgement may be obtained from explicit community ratings or via the endorsement of particular objects or persons by individuals (i.e. citation or recommendation networks).*

Table 4: Use of Reputation Systems in Online Enquiry Platforms.

Type	Platform	Vote to Promote	Content Rating and Ranking	Content Reviews and Comments	User Karma (Points)	Quality Karma	Abuse Reporting
Forum	vBulletin	+	+	+	+	+	+
	Jive Forums†		++		++	++	+
	Jive Engage	+	++	+	+++	++	+
	Lithium	+	++	+	+++	++	+
	Salesforce	++	+	+	+	++	+
Q&A	Yahoo! Ans.	++	+++	+++	+++	++	+
	Quora	++	++	++	+++	++	
	Stack Ex.	++	+++	++	+++	+++	+++

†Jive Forums is now replaced by Jive Engage.
Abbreviations: + = Yes. ++ = Multiple types. +++ = Extensively.

[32] Farmer and Glass (2010)

Reputation systems are built on *claims*.[32] Such *claims* form a relation between a *source* and a *target*. The *source*, usually a user makes a *reputation claim* about a particular *target* or object. Although *reputation* claims can be collected automatically, *reputation* systems tend to collect directly user feedback using particular user interfaces such as star ratings, thumbs up/down or votes. In the Q&A communities studied in this thesis, *best answers* can be identified by question askers as question solutions and individual answers can be rated positively if they are found helpful.

Besides allowing for the rating of answers, many other features help the identification of quality content and experts in Q&A communities. As shown in Table 4, recent Q&A platforms support different *reputation* features while systems not initially designed for managing enquiry communities tend to use less *reputation* attributes.

Reputation systems are perhaps the most reliable source of information concerning what is the most valuable content in a given community and can be seen as a measure of content quality. However, *reputation* information may not be always available as users may not vote for particular content or content may need time to obtain community ratings. As a consequence, *reputation* information cannot be used for all community content and needs to be derived through other means.

Besides the *reputation* of answerers and answers, many different features may be used for identifying valuable content. For instance, the length of answers or the lexical complexity of answers may be used for distinguishing quality answers from less accurate answers. In this thesis a large amount of such *metadata* is used as predictor of *best answers* (Chapter 4). A few of such metadata is discussed in more details in chapter 3 and many of them are proposed and used in chapter 4 when proposing an initial model for automatically identifying *best answers*.

Reputation information about users and answers as well as *non-reputation* features can be accessed for each answer to a question and be used for comparing answers within an answering thread. Since the structure of Q&A communities dictates that answers to a given question are bundled into answering threads and that only a unique answer can be identified as *best answer* for a given question, the relative value of *reputation* and answer features may be used for comparing the quality of answers within a thread.

As a summary, *reputation* information as well as other features may be used for comparing answers within a same thread in order to identify *best answers*. However, *reputation* may not be always

available and cannot be relied upon all the time as it may be missing. Nevertheless by coupling available metadata with the structure of answering thread, it is possible to create feature normalisation methods that compare the relative value of features between thread answers.

2.3.5 *Discussion and Structural Optimisation*

The thread-like structure of Q&A communities means that it is possible to consider only answers that are associated with a given question when trying to identify *best answers*. Such an approach differs from existing research that have mostly ignored such information by focusing principally on community-wide classifiers and disregarding the relations between same thread answers when trying to identify *best answers* (Chapter 3).

Since distinct answers tend to have different *reputation* and features and only a unique answer can be labelled as *best answer* within a thread, identifying the *best answer* of a given thread can be seen as the relative comparison of feature values between answers of a same question. For example, one might expect that the answerer with the highest *reputation* is more likely to provide the *best answer* compared to the other answerers that have less *reputation*. Similarly, the longest answer may be more likely to be associated with the *best answers*.

These observations motivate the design of feature normalisation methods that take into account the relative ranking between the

same features of different answers as well as the design of classification algorithms that use the thread wise nature of Q&A communities.

Therefore, as part of the proposed structural design methodology pursued in this thesis, two different structural optimisation methods are proposed for evaluating the pertinence of the structural design methodology introduced in this thesis.

First, after evaluating their feasibility on a small subset of features (Chapter 4), different systematic thread-wise feature normalisation methods are proposed (Chapter 5).

Second, a model optimisation approach that takes into account the thread-wise nature of Q&A communities is proposed (Chapter 5). The particular approach explored in this thesis relies on the usage of LTR models due to the similarities between INFORMATION RETRIEVAL (IR) tasks and *best answer* identification. Such a relation is discussed in details in Chapter 5 when the model optimisation approach is presented.

As a summary, in this thesis, two main different structural optimisation methods are designed and evaluated. Different structural optimisation methods based on thread-wise normalisation are proposed and an LTR algorithm is evaluated. Both models are presented and evaluated in Chapter 5.

2.4 STUDIES AND QUALITATIVE INDICATORS OF BEST ANSWERS

A common approach for identifying quality content is to use features-based automatic classifiers that try to distinguish *best answers* from *non-best answers* (Chapter 3). Existing research has mostly relied on the usage of intuition in order to select such features. One of the aims of this thesis is to evaluate if user studies and questionnaires can be used for better determination of what type of features can be designed for identifying *best answers*.

For better identifying what area to investigate when designing new features, the following section reviews existing surveys and user studies about what makes valuable communities and *best answers* in Q&A communities. Beside this review, a new survey is conducted on two business communities including the SCN community that is studied in this thesis for better understanding users' beliefs concerning what make *best answers*.

In the next subsections, the survey conducted for this thesis is presented before analysing different characteristics of Q&A communities and discussing the retained factors that are selected as part of the qualitative design methodology pursued in this thesis.

2.4.1 *Exploratory Community Survey*

Besides discussing existing survey and questionnaires about Q&A communities, a new study on two different business oriented communities was conducted in order to understand the factors that influence the creation of quality content and *best answers* Q&A communities.

The questionnaire was issued in the context of the RISK AND OPPORTUNITY MANAGEMENT OF HUGE-SCALE BUSINESS COMMUNITY COOPERATION (ROBUST) project[33] and also contained questions about the tools used by those communities. The two communities analysed, IBM Connections[34] and the SCN forums[35] are not limited to Q&A even though they are used mostly by users to get support and answers to their questions. In this context, all the questions asked by the questionnaire were not all relevant for the proposed research. Therefore, only the parts of the questionnaire related to community value and content quality are reported in the following sections.

[33] ROBUST, http://www.robust-project.eu/.

[34] *IBM Connections*, http://www-03.ibm.com/software/products/en/conn.

[35] SCN *forums*, http://scn.sap.com.

The 20 questions of the questionnaire were based on five point Likert-type scales of 1 to 5 with 1 representing *Never*, *Strongly disagree* and *Completely irrelevant*, 5 representing *Very Often*, *Strongly agree* and *Very important*. The questionnaire was sent to around 4000 users of IBM Lotus Connections communities and 40 selected users of the SCN community. The users were selected by the ROBUST project parters (SAP and IBM). For IBM, the questionnaire was sent to their internal user base while the SCN recipients were based on their community experience (e.g. community managers and high profile users). From such users, 197 responses were

obtained from IBM (151 fully completed questionnaires) covering 53 different sub-communities for the user questionnaire and 40 responses from SCN.[36]

[36] Rowe et al. (2011a)

2.4.2 Characteristics of Valuable Communities

Before analysing what are the factors that make *best answers*, the survey asked community users and managers to discuss the characteristics that make their community valuable and what makes users valuable. In the context of Q&A communities, community value may be intuitively defined as the ability of its users to answer each other questions successfully and promptly and valuable users are users that exhibit characteristics in line with valuable communities (i.e. answers to questions successfully and promptly). Such definitions relate to the idea that the sense of community is generally the result of a common goal and shared aims[37]. Formally, this thesis use the following definition of community value:

[37] Porter (2004)

Definition 2.8 (Community Value)**.** *Community value is defined as the ability of its members to reach a common and shared goal. In the case of* Q&A *communities, community value may be defined as the ability of a community to provide answers to questions successfully (i.e. providing* best answers*).*

Users and community managers were surveyed by proposing them to rate some statements concerning the value of their community in order to better understand what makes a community successful (Q9: "*The value of this community comes from...*"). The value of

the users that makes such communities was also surveyed (Q10: "*A valuable community member...*"). The questionnaire also asked users to rate their own value (Q11: "*Do you consider yourself a valuable member?*") and explain why they are valuable or not (Q12: "*If you answered 'yes' to the previous question, why? Or if you answered 'no', why not?*"). Finally, users were also asked to identify what are the features of stand-out users both positively (Q16: "*What features best describe interesting users?*") and negatively (Q17: "*What features best describe annoying or unhelpful users?*")

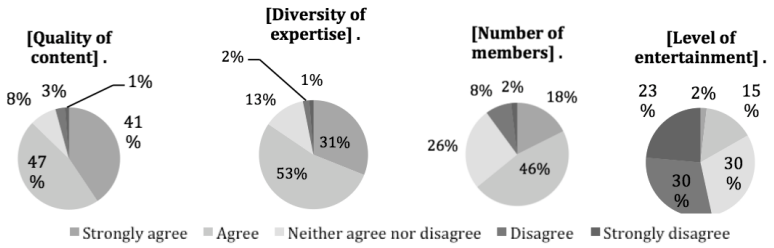


Figure 3: Perceived Community Value.

The results of *Q*9 (Figure 3 and Table 5) showed that the users of such communities consider *content quality* (88% *strong agree* or *agree*) as the most important measure of community value followed by the *diversity of expertise* (84% *strong agree* or *agree*). Not unsurprisingly, users generally discarded the *level of entertainment* as unimportant (53% *strong disagree* or *disagree*) meaning that most users seek knowledge and quality content answers instead of engaging in more social activities. Interestingly, it was observed that the average score obtained by all the statements is around 3.57. This result shows that users consider that community value is a an aggregation of many characteristics.

Table 5: Most valuable community characteristics ranked by average score (*max* = 5).

R.	*"The value of this community comes from..."*	Avg. Score
1	Quality of content.	4.3
2	Diversity of expertise.	4.1
3	Variety of topics and contributions.	4.0
4	Number of members.	3.7
5	Relationship between community members.	3.6
6	Quality of technical support.	3.6
7	Density of demographics.	3.5

The results of *Q*10 (Figure 4 and Table 6) showed that users that provide *accurate answers* are the most valuable (94% *strong agree* or *agree*) as well as users that provide *good quality and well presented content* (93% *strong agree* or *agree*). Such results show that the

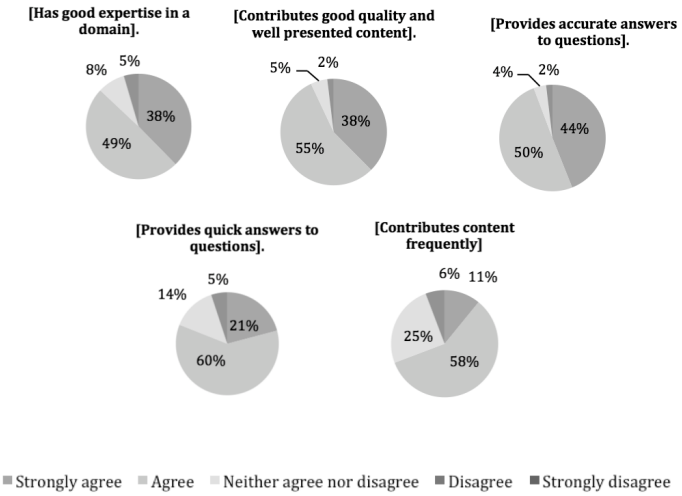


Figure 4: Most Important Attributes of Valuable Users in Enquiry Communities.

value of online communities come from the value of its individuals since valuable communities provide *quality content* and valuable users create *good quality* content. Beside the ability to provide quality content and answers, valuable users are *experts in a domain* (81% *strong agree* or *agree*) and answer promptly and frequently. As with the previous question it can be observed that sociability is not important (52% *strong disagree* or *disagree*).

Users were considering themselves valuable due to their experience, frequency of contributions and quality of their content while other users thought that they were not important because they are only interested in answers or are not experienced enough.

The outstanding users ($Q16$) were deemed important thanks to the *informativeness* of their contributions and *knowledge* (30% each) and *helpfulness* (24%) whereas invaluable users ($Q19$) are providing *inaccurate answers*, are *rude* and use *bad grammar*.

Besides the result of this study, a few other works have looked at the identification of what makes a community successful. Some studies have suggested that social interactions are generally important for

thriving communities as well as clear community policies [38] and highlighted the need of methods for measuring those attributes.[39] Mamykina et al.'[40] interviews also confirmed that a strong sense of purpose is required by successful Q&A communities. Another study by Brandtzæg and Heim[41] also suggested that community participation was also strongly associated with the presence of quality content. Compared to previous studies, the study presented in this thesis showed that users do not value as much social activities. Such a result is due to the particularities of Q&A since such websites are centred on information sharing rather than on user networking.

In general, this new study confirmed that valuable communities are highly associated with users that are able to *produce quality content*, have high *expertise* and are *responsive*. Such results suggest that the valuable communities can be measured by identifying the amount of quality content created and that the focus on *best answers* is highly relevant as for seeking to provide means for community manager to understand the value of their communities.

[38] *Preece (2001)*

[39] *Vatrapu et al. (2008)*

[40] *Mamykina et al. (2011)*

[41] *Brandtzæg and Heim (2007)*

Table 6: Most valuable user characteristics ranked by average score ($max = 5$).

R.	*"A valuable community member..."*	Avg. Score
1	Provides accurate answers to questions.	4.4
2	Contributes good quality and well presented content.	4.3
3	Has good expertise in an domain.	4.2
4	Provides quick answers to questions.	4.0
5	Contributes content frequently.	3.7
6	Has high ratings (i.e. *reputation*.)	3.4
7	Shares your interests	3.0.
8	Has many contacts (i.e. friends).	2.5
9	Has many fans (i.e. followers).	2.4

2.4.3 *Attributes of Best Answers*

Since the main aim of the user study is to determine what are the factors that are associated with *best answers*, the survey asked users to describe the factors that identify *best answers* (Q13: "*What factors influence your choice of* best answer*?*"). The results showed that users select *best answers* based on the *quality of the content* (36%), *clarity of the answer* (22%) and the *timeliness of the answer* as well as the *rating of answers* (both 14%). To some extent,

the users also take into account the *length of answers* (11%) but surprisingly only 1% of the answerers considered answerer *reputation* as important.

Similarly to previous observations, users seem to identify quality answers based on their intrinsic quality as well as their reactivity. To some extent users also rely on community ratings to help them to identifying quality answers.

Such results confirm and extend previous qualitative studies on content quality in Q&A communities.[42] For instance, Kim et al.[43] studied the justifications of users for picking *best answers* when they left comments. They analysed 457 *best answers* in Yahoo Answers and determined that quality is mostly related with *content values* (i.e. accuracy, clarity etc.). They also highlighted the importance of the *socio-emotional values* of answers (i.e answerer attitude, agreement, experience, etc.) as an important factor with opinion questions. The importance of such social factors is low in the survey conducted in this thesis as the communities studied are not opinion oriented. In their study Kim et al.,[44] found that for information questions, the importance of these factors is much lower compared to opinion answers. Kim et al. extended their study in a further publication[45] that confirmed their results while also analysing topical dynamics of quality criteria. In another survey, Fichman[46] took a small random sample of questions (1522) on 4 different communities including Yahoo Answers and annotated the *accuracy* (correct answer), *completeness* (whether an answer fully answers a question) and *verifiability* (the presence of external source) of answers. They found that *best answers* were more accurate but not necessarily as complete or verifiable as other types of answers. Nevertheless,

[42] *Kim et al. (2007); Fichman (2011)*

[43] *Kim et al. (2007)*

[44] *Kim et al. (2007)*

[45] *Kim and Oh (2009)*

[46] *Fichman (2011)*

the results on *best answers* showed that *best answers* were mostly complete (on average 71%), accurate (on average 43.5%) and verifiable (on average 24%) confirming our findings on the importance of quality and accuracy for *best answer* identification.

2.4.4 Contributors Motivations

The questionnaire conducted for this thesis also looked at user motivations by asking users why they get involved in online communities (Q14: "*Why do you participate in this online community?*"). Compared with previous questions that directly looked at the value of users and communities, these questions give some insights concerning the behaviour of users in Q&A communities and what activities they consider important. Such information may be used as an indicator of what behaviour is associated with valuable users and by extension quality content and *best answers* (Figure 5 and Table 7).

Table 7: Most cited reasons for contributions ranked by average score ($max = 5$).

R.	"*Why do you participate?*"	Avg. Score
1	To learn new things.	4.4
2	To help people with their problems.	3.7
3	To draw attention to your own business.	3.6
4	To participate in discussion.	3.6
5	To share opinions and ideas.	3.6
6	To network with people.	3.3
7	To be recognised.	2.6
8	To form new groups and supporters to a cause or event	2.4

Users generally stated that they get involved in order to *learn new things* (96% *strong agree* or *agree*) and to *help people with their problems* (71% *strong agree* or *agree*). Such results are also corroborating previous questions results as users value the *knowledge* of others and aim at *providing accurate and good answers*. Unsurprisingly, users are not interested in having fun (63% *strong disagree* or *disagree*) as they only get involved in order to get answers to their questions.

Similarly to the previous insights, the findings about user motivations are similar to previous surveys.[47] For example, Nam et al.[48]

[47] Nam et al. (2009); Mamykina et al. (2011)

[48] Nam et al. (2009)

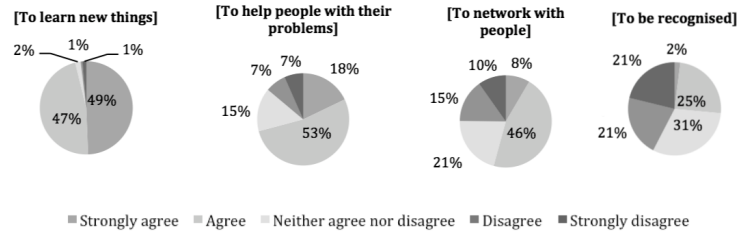


Figure 5: Most important community participation factors in Enquiry Communities.

analysed and interviewed some members of the Naver community[49] and highlighted the importance for contributors of learning new things and helping others confirming our findings. Using a relatively similar interview approach, Mamykina et al.[50] asked six highly ranked contributors of the Stack Overflow website about their motivation for contributing. They found that intrinsic motivational factors where important such as the desire to help their community and learn new things. Extrinsic motivations was also present with, for example, a wish to enrich their professional portfolios or collect *reputation* points[51]. Another more general work by Raban and Harper[52] reviewed existing work on user motivations in online communities and found that users were motivated by their perception of the value of a community such as the amount of knowledge held by participants and the similarities between themselves and the community they are participating in.

From the thesis survey it can be observed that user are mostly motivated by obtaining more knowledge about a particular topic or helping less knowledgeable users. This result shows that the value of a community mostly depends on the ability to obtain quality answers that solve their issues (i.e. *best answers*) or have means for asking questions.

[49] *Naver Knowledge iN*, http://kin.naver.com.

[50] Mamykina et al. (2011)

[51] Mamykina et al. (2011)

[52] Raban and Harper (2008)

2.4.5 *Discussion and Qualitative Features*

Previous studies and the new user survey conducted in this thesis identified different areas that users consider related to *best answers* and valuable communities. In general, the new findings are in line with previous surveys.[53] The results show that *best answers* (Section 2.4.3) and content quality share a lot in common (e.g. accuracy, clarity). As a consequence, it can be argued that analysing *best answers* is similar to identifying quality answer in Q&A communities. This observation suggests that the intrinsic quality of answers is more than just a constituent of *best answers* and cannot be easily measured as an independent predictor of *best answer*.

[53] *Preece (2001); Vatrapu et al. (2008); Mamykina et al. (2011); Brandtzæg and Heim (2007); Kim et al. (2007); Fichman (2011); Nam et al. (2009); Raban and Harper (2008)*

Although the new user survey only received 151 full responses out of the 4000 contacted users for the IBM user sample, the surveyed user base was much bigger than in the previous studies were the mean user sample size was 80.[54] Similarly, even if the SCN user base was relatively small, the users that were selected were explicitly picked for their involvement in the community. Therefore, the surveyed SCN users were knowledgeable contributors and more likely to know well the needs and requirement of their community. As a consequence the experience of the small SCN user sample is likely to make up from the shortcomings of having a small user survey.

[54] *Vatrapu et al. (2008); Mamykina et al. (2011); Brandtzæg and Heim (2007); Nam et al. (2009)*

Despite the previous shortcomings, the presented survey gives insights concerning user motivations and what they consider valuable. This study showed that the value of a community is highly related to the ability of its users to produce content according to particular characteristics, namely quality content and good answers.

The survey findings showed that valuable communities are associated with users that are able to *produce quality content*, have high *expertise* and are *responsive*. Similarly, contributors' motivations where centred on their willingness to be *knowledgeable* (i.e. *learn new things*) and *helping other users*. For the factors that are associated with *best answers*, the thesis survey identified *quality and clarity of answers* as well as *community reactivity* (i.e. timeliness of the answer) and *answers ratings* as important indicators of *best answers*.

Although many different factors are worth investigating, in this thesis, the factors that are associated with *best answers* that are investigated are: 1) The quality and clarity of the answers created by users; 2) The expertise and ability of answerer to learn new things and being knowledgeable, and; 3) The answering reactivity of users (i.e. timelessness).

Rather than modelling expertise directly by only relying on community ratings, this thesis proposes to learn expertise as the *ability of learning new things*. The proposed approach is to model the *complexity* (i.e. difficulty) of questions and use it for representing the *maturity* of users as their ability to answer more complex questions over time. It is decided to focus on questions complexity rather than answer complexity as the complexity of answer is likely to be dependent on question complexity (i.e. users that reply to complex questions are likely to provide complex answers) whereas the opposite may not be true.

The advantages of this approach is that it takes the answering behaviour of answerers by taking into account the *complexity* of questions as well as answerers' ability to improve their answering skills

overtime. Existing approaches related to question *complexity* and user *maturity* are discussed in Chapter 3 while the proposed models of question *complexity* and *maturity* are discussed and evaluated in Chapter 6.

Similarly to the previous approach, instead of only modelling the reactivity of answerers based on time-to-answer information, this thesis investigates the implicit modelling of the amount of work or *effort* that users put into their contributions. Basically, the proposed approach aims at representing the amount of time a user needs for producing a particular answer. As with the modelling of user *maturity*, this approach has the advantage of modelling the latent behaviour that triggers an observed response time for a particular answer, thus making it potentially more accurate than response times. Existing approaches related to contribution *effort* are discussed in Chapter 3 while the proposed model of contribution *effort* is discussed and evaluated in Chapter 7.

As a summary, in this thesis, two features based on qualitative analysis are designed and evaluated. First, the concept of user maturity is proposed as a proxy measure of user knowledge and ability to learn new things. Secondly, different contribution effort models are proposed for representing the reactivity of answerers. Related work is discussed in the following chapter whereas each of such models are introduced in Chapter 7 and Chapter 6 respectively.

2.5 Experimental Approaches and Evaluation Methods

Depending on the experiment, different methods can be applied for modelling experiments. The previous studies show that the automatic identification of *best answers* can be achieved using binary classification models and that the modelling of user maturity and contribution effort may improve *best answers* identification models.

The following sections introduce some background concerning the modelling approaches, evaluation methods and measures used for conducting experiments related with *best answer* identification and the representation of user maturity and effort.

Most experiments conducted in this thesis follow a standard experimental approach by first creating a prediction model that is then evaluated using a set of evaluation metrics. The results are then studied in detail using feature analysis techniques and optimised using the insights from such analysis.

The following sections give some background on the different modelling approaches used in this thesis and the literature. The merits of different evaluation measures are discussed as well as the methods used for performing analysing models and optimising them.

2.5.1 *Modelling Approaches*

The types of models used for identifying best answers and representing user maturity and effort differ greatly. In the case of best answers identification, the goal is to classify documents whereas the aim of maturity and effort modelling is to create predictors that are incorporated in classification models.

Binary Classifiers: The goal of a binary classifiers is to classify document instances as belonging to a class or otherwise. In the case of automatic *best answers* identification, the aim is to classify answers as *best answers* or *non-best answers*.

A wide variety of approaches exist and are used in different situations depending on the type of input and requirement of a particular model. For example, decision trees approaches such as C4.5/J48[55] use information entropy to chose how to split tree node when building a prediction model. Tree based models are particularly useful when results need to be interpreted easily. In more complex cases, other models such as SUPPORT VECTOR MACHINES (SVM)[56] may be used when complex relations exist between input variables. However such type of model are generally more complex to interpret.

[55] *Quinlan (1993)*

[56] *Cortes et al. (1995)*

The literature differentiate supervised and unsupervised models as well as semi-supervised approaches. In the case of supervised models a certain amount of already annotated data is provided for a model to learn. In an unsupervised setting, no labels are. In this case the classification problem can be seen as a clustering task where the goal is to group a set of input documents based on their similarity.

Finally, semi-supervised methods use partially labelled data in order to label new data.

Although classification models may be supervised or unsupervised, this thesis focus on supervised models as *best answers* annotations are available in the datasets studied. The experiments conducted in this thesis use the ALTERNATING DECISION TREE (ADTREE) algorithm,[57] a type of decision tree algorithm, as it gives good results for classification tasks.

[57] *Freund and Freund (1999)*

Ranking Models: Ranking models also called LTR models are designed for learning the order of a set of documents based on relevance labels. Compared with classification models, they usually take into account whole document sets during the prediction task. Such models have been historically used in IR for ranking search query outcomes.[58]

[58] *Liu (2009)*

LTR approaches are divided in three different categories 25: 1) pointwise methods; 2) pairwise models, and; 3) listwise approaches. The pointwise approach is based on the classification of single documents. Each document is directly evaluated on a given ranking function and an absolute relevance score is returned that gives the relevance and absolute position of a document. The pairwise approach does not assume absolute relevance labels but instead focus on the comparison of document pairs. Documents are ranked according to their preference order score obtained from a ranking function that compare document pairs. The listwise approach directly treat document lists as learning instances and learn a ranking function that directly return ranked lists rather than individual rank for each list documents. Therefore, instead of reducing ranking as a

classification task, learning is achieved directly on document lists: an entire ranked list is treated as a learning instance.

LTR models can be used for identifying *best answers* as identifying a *best answer* in an answering thread is similar to ranking answers by *best answer* likelihood. The method used for applying LTR models for identifying *best answers* is discussed in Chapter 5.

Regression Models: Regression models are statistical methods for estimating the relationships between variables. They are different to standard classification models as they can also infer continuous values from different input variables.

The most famous regression model is the linear regression, a method that estimates the linear relationships between variables in order to predict a given outcome. Another important method is the logistic regression, a similar approach with the ability to infer binary classes.

Regression models are used in previous research related to *best answers* identification as well as in Chapter 6 where it is applied for modelling the complexity of questions and the maturity of users.

Probabilistic Topic Models: Probabilistic topic models are probabilistic models that infer the content of documents by identifying their topics where each topic is represented by distribution of words. Topic models are generative models as they try to estimate latent variables that have generated the words of a document.[59]

[59] *Steyvers and Griffiths*

One of the most popular topic modelling approach, called LATENT DIRICHLET ALLOCATION (LDA) has been proposed by Blei et al.

and models topics as a probabilistic distribution of words and documents as a distribution of topics. Such generative model estimates word-topic and document-topic probability distributions.

Many extensions have been proposed and applied for different use case. For example, the DYNAMIC TOPIC MODEL (DTM)[60] was proposed for tracking the evolution of topics over time. AUTHOR TOPIC MODEL (ATM) was created for associating topics to document authors and the JOINT SENTIMENT TOPIC MODEL (JST)[61] was created for representing the sentiment associated with documents and topics.

[60] *Blei and Lafferty (2006)*

[61] *Lin and He (2009)*

In general, topic models need little or no supervision. However, their main weakness is that they require a large amount of computations and tend to use indirect evaluation measures.

In this thesis a model extending the LDA, JST and ATM models is described in Chapter 7 for modelling the amount of labour required by users to answer questions over time.

Graph Measures and Metrics: Many of the previous models use different set features for performing predictions and are based on some simple metrics that can be directly measured from the data. For example, in Q&A communities, some of such measures can be *answer length*, *user reputation* and *answer number*.

A particular type of feature that needs more computation is generated from graph measures that use the network structure of the analysed data. For example, in Q&A, such type of measure can rely on the connection between asker and answers through their questions and answers for deriving a particular measure.

Perhaps one of the most know graph algorithm is Page Rank.[62]

[62] *Page et al. (1999)*

Page Rank was initially designed for ranking websites based on their hyperlink network. The idea is that important websites are referenced by many websites and are associated with high weight. Then, the algorithm computes such weights through the network so that some links are more important that others. As a results, websites that are linked from other important website are given high ranking. Another similar measure is HYPERLINK-INDUCED TOPIC SEARCH (HITS)[63], however HITS distinguishes two type of websites: hub websites that cite many other pages (e.g. web directories), and, authority websites that are highly cited.

[63] *Kleinberg (1999b)*

Other common measures include simple measures that calculate the number of nodes or edges in a network or degree measures that calculate the number of edges linked to a given node.

For more examples of graph measures, see: `https://reference.wolfram.com/language/guide/GraphMeasures.html`.

In the context of *best answers* identification and related work, graph measures are primarily used for propagating the reputations of users using the asker answerers connections based on the questions and answers they reply and ask.

Although graph measures have the advantage to enable the computation of complex measures such as the reputation of users, they are computationally intensive. As a consequence, this thesis focus on more standard measures as they tend to be more widely used in practice. Moreover, as it can be observed from the literature (Chapter 3), most graph metrics are used for reputation propagation. Since the communities studied in this thesis already have reputation measures, it is not really necessary to use reputation propagation methods.

2.5.2 *Evaluation Methods*

Supervised binary classification models are typically evaluated using two annotated datasets that contains positive and negative instances. First a training set is used for building a model using a particular ML algorithm. Then the model is evaluated on an held-out dataset using different evaluation measures for assessing its performance (Section 2.5.3).

Different methods exists for comparing results such as splitting a dataset in two parts where 80% of the data is used for training purposes and the remaining 20% are kept for evaluation.

A better approach is to perform k-folds cross-validation. The idea is to divide a dataset into k subsets and then keeping one subset for testing and the other k-1 subsets for training. The process is repeated k times and the results are averaged.

An additional improvement over k-folds cross-validation is stratified k-fold cross-validation where the folds are generated so that value is approximately equal in all the folds. In the case of a binary classifier, this means that each fold contains roughly the same proportions of positive and negative class labels.

In this thesis, 10-folds cross-validation is used.

2.5.3 *Evaluation Measures*

Depending on the type of experiment, different evaluation metrics can be used for analysing the accuracy of findings. Binary classification models generally rely on four different measures that indicates some accuracy aspect of a given classifier.

Precision: In a binary classification task, precision (P) is the fraction of correctly classified documents returned by a classifier and is calculated from the true positive (tp) and false positive (fp) values that respectively represent the number of positive documents successfully classified and the negative documents wrongly classified as positive instances:

$$P = \frac{tp}{tp + fp} \tag{1}$$

In the context of *best answer* identification, the precision measure represents the proportion of retrieved *best answers* that are real *best answers*. It shows how *best answers* are successfully classified compared to misclassified answers. The precision measure is useful when the main goal of a model is to not misclassify instances.

Recall: Recall (R) represents the fraction of positive documents that are returned by a binary classifier. It is calculated from the true positive (tp) and the false negative (fn) values. The false negative

value corresponds to the number of positive documents wrongly classified as negative instances:

$$R = \frac{tp}{tp + fn} \tag{2}$$

In the context of *best answer* identification, recall measures the proportion of *best answers* that are successfully retrieved. The recall measure is important when the main goal of a model is to make sure that positive instances are not missed.

AREA UNDER THE CURVE (AUC): The AUC is based on a graphical plot called the RECEIVER OPERATING CHARACTERISTIC (ROC) curve that represents the performance of a binary classifier model as its discrimination threshold is modified. The curve is created by plotting the TRUE POSITIVE RATE (TPR) against the FALSE POSITIVE RATE (FPR) at various threshold settings. The TPR and FPR measured using the true positive (tp) and false positive (fp) values at different discrimination threshold settings.[64]

[64] *Ling et al. (2003)*

The AUC value focus on the precision aspect of a given model and is useful for determining the overall performance of positive classification in binary classifiers.

F-Measure: The F-measure or F_1 score combines both precision P and recall R using their harmonic mean. The F_1 score is calculated using the following equation:

$$F_1 = 2 \times \frac{P \times R}{P + R} \tag{3}$$

The F_1 score has the advantage to balance the importance of both the precision and recall scores. As a result it can be seen as a good measure for assessing the general performance of a model.

In this thesis the three previous measures are computed for each experiment but the analysis focus is on the F_1 measure as it is a balanced metric that represents both the recall and precision of *best answers* predictions.

Perplexity: The perplexity measure is used for determining the ability of a probabilistic model to predict unseen data. It is commonly used in topic modelling for comparing different models and is defined as the reciprocal geometric mean of the likelihood of a test corpus given a trained model's Markov chain state $\mathcal{M}$. A lower perplexity means a better perfectiveness, and therefore a better model. For a topic model such as LDA, the perplexity is defined as the per-word perplexity of the unseen test set $\widetilde{\mathcal{D}} = \{\widetilde{w}_d\}_{d=1}^{D}$ based on the previously trained model $\mathcal{M} = w, k', k, e$. $\widetilde{w}_d$ represents the word vector of the dth document in the test set, $\widetilde{C}_d$ is the total number of words in $\widetilde{w}_d$ and the perplexity of unseen documents is defined as:

$$Perplexity = P(\widetilde{\mathcal{D}} \mid \mathcal{M}) = \exp\left\{ -\frac{\sum_{d=1}^{\mathcal{D}} \log P(\widetilde{w}_d mid \mathcal{M})}{\sum_{d=1}^{\mathcal{D}} \widetilde{C}_d} \right\} \quad (4)$$

The perplexity measure is used in Chapter 7 for evaluating the comparison of different topic models.

T-test: A t-test is a statistical test for determining if two sets of data are significantly different from each other. The idea is to determine if a given variable mean is close to a given value and decide if the compared distributions are similar. The assumption is that if both set of values are similar, they should follow a similar distribution. A p-value measure is used for indicating if two sets are significantly different. By comparing the p-value to predetermined confidence intervals, it can be determined if the the two sets are significantly different.

Depending on the analysis t-test can be paired and tailed. A paired t-test is used when the same measurements are conduced in different settings. A tailed t-test can be used for testing the significance in a given direction. For example, a tailed t-test can be applied for studying the relation of answer length and *best answers* in order to determine if a *best answer* is more likely to be associated with small answers or long answers. In this case two separate tailed test are performed for identifying these relations. First a left handed t-test is done for checking if *best answers* are likely to be linked with short answers. Then, a right handed test is done for checking the opposite relation. Depending on the significance value, it can be decided if *best answers* are associated with short or long answers.

t-tests are used in this thesis in many chapter for determining the significance of results. This approach is used extensively in Chapter 7 for confirming that the developed models can be used for modelling contribution effort.

Kappa Statistic: When dealing with user annotations, it is necessary to identify if the different annotators have an agreement concerning the annotation of the same items in order to decide if the annotations are valid. Inter-rater agreement can be calculated using Fleiss' kappa measure.[65]. The kappa measure is normalised between 0 and 1 and the agreement is total when the value reach 1.

[65] Gwet (2001)

A full explanation concerning the calculation of the kappa measure and alternatives can be found in "Handbook of inter-rater reliability" by Kilem L. Gwet.

2.5.4 *Features Analysis and Model Optimisation*

After training a particular feature-based model, it is useful to analyse the relevance of its features for obtaining insight about how each feature influences a model. Feature influence can also guide model improvements by removing non relevant predictors. For instance, in the case of *best answer* prediction, feature analysis can give insights like if *answer length* correlate with *best answers* and if long answers are more likely to be *best answers*. Similarly, if it is found that a particular predictor does not correlate, it can be removed from a feature-based model for increasing its accuracy.

In this thesis and in the literature, different methods are used for analysing the importance of features The usual approach is to perform feature ranking based on a score that is given by a feature importance metric Then, features can be added in a new model progressively until improvement in precision, recall, AUC or F_1 is maximised.

Visual methods and t-tests can be also used for determining how a feature affects a particular classification model. For example box plots can visually represent variable values distribution for different

data groups (e.g. *best answers* and *non-best answers*) and used for determining how a particular value is associated with a given group. In order to determine if an association is significant t-tests can be performed in order to know if a lower or higher value of a particular variable is associated a given group.

Ablation Test: The ablation test or feature drop method is a simple approach for estimating the importance of individual features by reporting the decrease of accuracy in a model when a particular feature is removed. For example, if the evaluation measure is F_1, a given model is trained first with all the features and then iteratively with the same features except one. By reporting the difference between the full model F_1 and the newly obtained F_1 value, the importance of a particular predictor can be derived: a small F_1 difference indicates an unimportant feature while a high difference indicates high relevance.

INFORMATION GAIN (IG) and INFORMATION GAIN RATIO (IGR): IG and IGR are measures used in decision tree models for deciding what variable to pick for creating tree branches in order to maximise classification output. IG is based on the notion of entropy, a measure of event uncertainty noted $H(X)$ and conditional entropy, a measure that calculates the uncertainty of a particular event X conditionally to another event or observation Y noted $H(X|Y)$.

The IG formula calculates the decrease of entropy when a feature is present or not.[66] As a consequence, the higher the value the more

A full explanation concerning the calculation of the entropy measures can be found at the following address: `http://www.cims.nyu.edu/~chou/notes/infotheory.pdf`.

[66] Forman (2003)

important the variable. The IG formula $IG(X,Y)$ is defined as follows and calculates the IG associated with variable Y:

$$IG(X,Y) = H(X) - H(X|Y) \tag{5}$$

IGR is based on the IG but is calculated based on the ratio between the IG and the split information value, a measure that is used for normalising the IG associated with a given variable.[67]

[67] *Bramer (2013)*

In the context of feature ranking both IG and IGR are used for determining the decrease in entropy of a particular feature when trying to predict a given outcome (e.g. entropy decrease of not using *answer length* for identifying *best answers*). IGR is generally preferred as it less sensible to bias thanks to split information normalisation.

Correlation Feature Selection (CFS): CORRELATION FEATURE SELECTION (CFS) is a method designed for selecting features that are relevant to a particular model based on the hypothesis that "Good feature subsets contain features highly correlated with the classification, yet uncorrelated to each other".[68]

[68] *Hall (2000)*

In the context of feature ranking, CFS has the advantage of trying to rank features that are independent with each other but correlated with the target class. This particularity can be useful for producing more accurate classification models.

2.6 Analysed Communities and Datasets

The research presented in this thesis is performed on three different datasets extracted from three distinct communities. We picked communities that vary in structure, size and topics in order to validate our research on a wide range of communities. The summary of our datasets is given in Table 8.

Table 8: Datasets statistics for the SCN forums, SF and Cooking (CO).

	Statistics				
Dataset	Start Date	End Date	Users	Questions	Answers
SCN	12/2003	07/2011	32 942	95 015	427 221
SF	08/2008	03/2011	51 727	71 962	162 401
CO	07/2010	03/2011	4941	3065	9820

2.6.1 *SAP Community Network*

The SAP Community Network (SCN) is a set of forums designated for supporting SAP customers and developers. SCN integrates traditional Q&A functionalities systems such as *best answer* selection, user *reputation* and moderation. Each SCN thread is initiated with a question and each answer in that thread is a reply to that question. Thread authors can assign a limited number of points to the answers they like (unlimited two-points for *helpful answers*, two sets of six-points for *very helpful answers* and one ten-points for the *best answer*). Points given to answers add to the *reputation* of their authors. Users can be flagged as topic experts, get promoted to moderators, or be invited to particular SAP events if their online *reputation* is high.

The dataset consists of 95,015 threads and 427,221 posts divided between 32,942 users collected from 33 different forums between December 2003 and July 2011. Within those threads, 29,960 (32%) questions have *best answers*.

2.6.2 *Server Fault*

SERVER FAULT (SF) (Figure 6) is Q&A community of IT support professionals and is hosted on the STACK EXCHANGE (SE) platform. SE provides social features such as *voting*, *reputation* and *best answer selection* while making sure that each posted answer is self-contained. However, SF differences reside in its rewarding program where each user gains access to additional features like ability to vote and advertising removal depending on their *reputation*. Compared to SCN, SF editing policy is completely community driven. Depending on the user *reputation*, each community member is allowed to refine other people's questions and answers. Hence, instead of adding additional posts for elaborating questions or answers, SF users can directly edit existing content. To keep the community engaged, the SF platform offers rewards and badges for various type of contributions. For example, users can earn the *Autobiographer* badge if they fill their profiles completely. SF users' *reputation* is calculated from the votes that have been cast on a particular question or answer. For each post, community members vote up or down depending on the quality and usefulness that is then pushed to the post owner. As community members gain/lose *reputation*, they gain/lose particular levels and abilities.

The SF dataset is extracted from the April 2011 public dataset, and

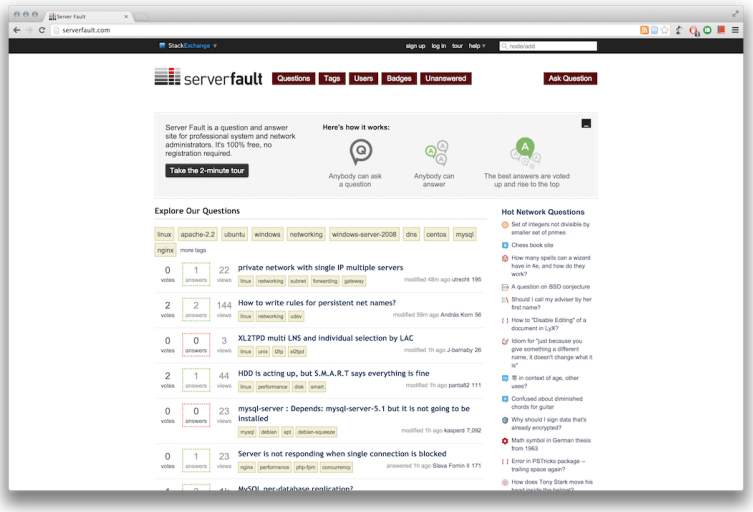


Figure 6: Picture of the SF community homepage.

consists of 71,962 questions, 162,401 answers and 51,727 users. Within those questions 36,717 (51%) questions have *best answers*.

2.6.3 *Cooking Website*

The COOKING (CO) community (Figure 7) is composed of enthusiasts seeking cooking advice and recipes. It is also hosted on the SE platform and thus share the same attributes and functionalities as SF above.

This dataset is smaller than the others with 3,065 questions, 9,820 answers and 4,941 users. The datasets contains 2,154 (70%) questions that have *best answers*.

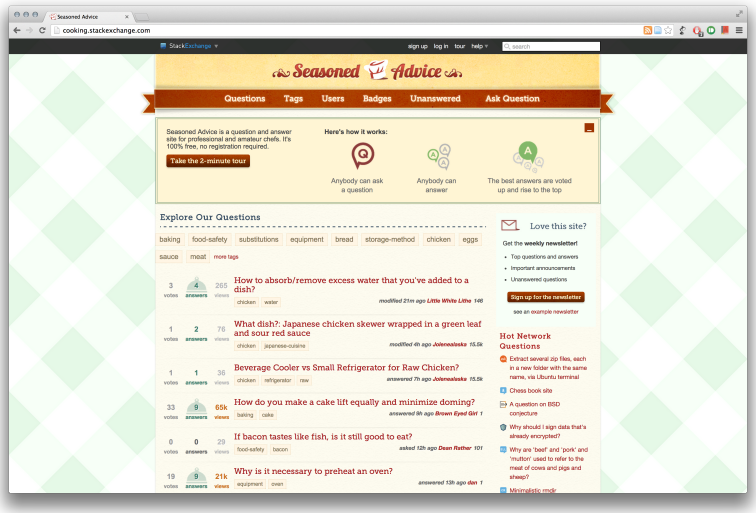


Figure 7: Picture of the CO community homepage.

2.7 SUMMARY

This chapter introduced the structural and qualitative design methodology used in this thesis for improving the identification of *best answers* in Q&A communities. First, the structure of Q&A websites was investigated in order to identify what type of structural optimisation can be designed. Second, a user survey was conducted and compared to previous studies in order to determine the factors that are associated with *best answers*.

Based on the structural analysis, two different optimisation methods were highlighted: 1) The usage of thread-wise normalisation methods, and; 2) The application of optimised algorithms based on LTR models. The qualitative survey identified many factors associated with quality answers and the following features were proposed based on the survey: 1) A model for representing question complexity and user maturity for identifying knowledgeable users, and; 2) A model of contribution effort for measuring the reactivity of users.

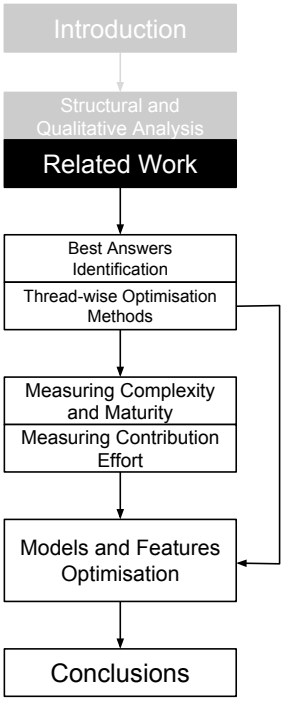
Besides the structural and qualitative analysis, this chapter discussed the methods used for performing and evaluating the experiments conducted in this thesis.

This chapter also introduced the three different communities and datasets studied in this thesis: 1) The *Cooking* (CO) community; 2) The SF dataset, and; 3) The SCN forums.

In the following chapter (Chapter 3), current work is reviewed in relation with *best answer* identification. In particular, present methods are reviewed for identifying *best answer* and measuring content *clarity*, contribution *effort*, question *complexity* and user *maturity* as their relation with *best answers* has been identified through the *qualitative design* methodology introduced in this thesis.

3 RELATED WORK

Structural optimisations that use the thread-like structure of Q&A communities may be used for creating more accurate models for automatically identifying *best answers* (Chapter 2). Similarly, well designed features can be used for the same purpose. In this context, features that model the quality of answers, the ability of users to answer complex questions (i.e. *maturity*) and the amount of *effort* that users put in their answers may help the identification of *best answers* (Chapter 2).



In this chapter existing research on *best answer* identification is reviewed as well as existing research relating to the design of models of content quality, question *complexity* and community *maturity* and contribution *effort*. This chapter also analyse the different optimisation methods used in previous works.

Based on the analysis of existing works, it can be observed that structural optimisation methods and qualitative approaches for modelling features have been largely ignored by previous work even thought different optimisations have been successfully applied to unrelated tasks. Existing work also highlights the accuracy of decision trees as well as the importance of using a large variety of features for the automatic identification of *best answers*. Based on

such observation, the baseline model used in this thesis uses a decision tree model and a large set of features. Concerning the creation of question complexity, maturity and contribution effort, existing works have mostly looked as related fields rather than tackling these particular features.

This chapter is divided as follows. First, the general approaches that have been used for identifying *best answers* and quality content in the past are discussed. This section is also used for reviewing structural optimisation methods that have been used within the existing models. Then, in the following section, existing research relating to the qualitative features retained in this thesis are investigated. In the third section, the limitations of existing approaches concerning *best answers* identification and qualitative features are highlighted as well as the differences with the work conducted in this thesis. Finally, the last section summarises the findings of the chapter.

3.1 INTRODUCTION

Following the previous chapter (Chapter 2), existing works on the automatic identification of quality and *best answers* is investigated and approaches related to *qualitative* and *structural design* are studied (RQ 1).

Although, much research has looked at identifying quality content and *best answers* in Q&A communities, existing research has largely ignored the benefits of using structural optimisation methods and

the usage of question *complexity*, community *maturity* and contribution *effort* features for improving the automatic identification of *best answers*.

In this thesis, user studies results are combined with the structural analysis of Q&A communities in order to pre-identify *best answer* factors and guide the design of *best answer* identification models by focusing on the design of metrics that represent such factors (RQ 1.2) and on structural optimisation methods (RQ 1.1). Although many factors can be used for building identification models this thesis focus on two factors (Chapter 2): 1) Question complexity and user maturity, and; 2) Contribution effort. Existing work in modelling such metrics is studied in detail in this chapter as well as research on content quality.

While reviewing previous work on quality content and automatic *best answer* identification, the approaches used for optimising such models are also reviewed as well as previously used feature normalisation methods.

As a summary, the contributions of this chapter are as follows:

- Review existing approaches and models for identifying quality content and *best answers* in Q&A communities and the different optimisations that have been previously used.
- Discuss research for measuring content quality, answer accuracy and clarity.
- Present work related to question complexity and community maturity.
- Review work on contribution effort modelling.

– Highlight the limitations of previous works and the difference with the research investigated in this thesis.

3.2 Best Answers Identification

The identification of quality content in online Q&A platforms has been focused on characterising good questions and good answers. In the area of quality answers, research has explored the identification of *best answers* and quality answers. Quality answers are generally identified as answers that are valuable given a set of quality criteria whereas *best answers* are answers that are labelled as solution from question owners. Although the focus of this thesis is on the identification of *best answers* and the application of qualitative and structural design (RQ 1), existing research on quality answer identification is also reviewed.

3.2.1 Best-Answer Identification

Work on *best answer* identification relates to the finding of the most suitable answer for a given question. However, it is distinct from automatic question answering[69] where Natural Language Processing (NLP) techniques are used for generating answers automatically.

[69] *Hirschman and Gaizauskas (2001)*

Some works have directly investigated the identification of *best answers*.[70] Most of the existing works have used feature based models in order to identify *best answers*. In general the literature distinguishes three different type of features: 1) Textual features derived

[70] *Adamic et al. (2008)*; *Blooma et al. (2008, 2012)*; *Tian et al. (2013)*; *Gkotsis et al. (2014)*

from the content of answer; 2) Non textual features derived from post metadata such as ratings, and; 3) Contextual or relational features derived by comparing answers or questions features within a thread. In this thesis features are distinguished by scope and therefore distinguish: 1) User features that represent the characteristics of authors of questions and answers; 2) Content features that represent the attributes of questions and answers, and; 3) Thread features that represent relations between answers in a particular answering thread.

In their paper, Adamic et al.[71] analysed the Yahoo Answer community to study if different features can predict *best answers* observed from poster ratings. Additional analyses were performed using different graph measures such as degree distribution, ego network analysis (from 100 sampled users), strongly connected component analysis and motif analysis in order to better understand user interactions in Q&A communities. The authors developed a topic entropy measure that calculates whether user posts are concentrated in a given main category. Their results showed that users that focus their answers on particular topics are more likely to be correlated with *best answers*. This suggests that topical focus may be useful for the analysis of the communities studied in this thesis as the studied content is mostly factual and technical. According to their study of *best answer* quality, long answers are positively correlated with *best answers* as well as the number of previous *best answers* for a user answering a particular question. However, it appears that longer threads are a sign of non *best answer* and a high number of answers for a given user is also associated with *non-best answers*. In general results were relatively poor with 62% accuracy and a logistic regression model.

[71] *Adamic et al. (2008)*

The usage of different features for identifying *best answers* has also been investigated by Blooma et al.[72]. In their paper, the authors explored the usage of non-textual features such as asker and answerer reputation and authority (i.e. number of *best answers*) and textual features like accuracy, completeness, language (i.e. spelling and grammar), reasonableness (i.e. consistency and believability) and content length. In order to perform their study, manual annotations of most of their textual features were required making their approach difficult to perform on large scale communities (Their analysis only focused on 300 questions-answers pairs). By analysing the coefficient returned by a trained regression model, they show that textual features are more important than non textual features. In particular, they show that completeness and answer length are the most important factors of *best answers*. In a set of extended studies[73], the authors added additional factors from other research [74] in order to increase their ability to identify *best answers* and better understand what factors are the most important. The results shows that ratings are good predictors of *best answers* showing that community ratings improve the identification of *best answers* even though they require community annotations that may not be available when trying to label answers as best or not *best answers*. Similarly to their previous work, they focused on a small dataset but extended it to 2400 answers.

[72] *Blooma et al. (2008)*

[73] *Blooma et al. (2012)*; *John et al. (2011)*

[74] *Gazan (2006)*; *Agichtein et al. (2008)*; *Jeon et al. (2006)*; *Bian et al. (2008)*; *Liu et al. (2008)*; *Sun et al. (2009)*

More recently, some work has explored the usage of relational or contextual features for identifying *best answers* in SO confirming the intuition presented in Chapter 4[75]. The work shows that relations between answers of a thread benefit the identification of *best answers* as they help separating *best answers* from standard answers. They used cosine similarity metrics between answers and

[75] *Tian et al. (2013)*

question and answers of a same thread and showed that the minimum similarity between answers as well as the number of concurrent answers helps the identification of *best answers*. They obtained good accuracy (72.3%) using the random forests algorithm.[76]

[76] Tian et al. (2013)

In this thesis a similar intuition is applied when developing the thread features (Chapter 4) that are used as part of the structural design methodology advocated in this thesis (RQ 1.1).

Building on the work presented in Chapter 4 and the idea of thread features, Gkotsis et al.[77] normalised textual features using their ranking within a thread (e.g. they replaced the answer length features by a discrete number corresponding to their relative length within a thread.). The conducted analysis was performed on multiple datasets from the SE websites and they obtained good results ($F_1 = 0.76$) when only using normalised textual features and alternating decision trees. In this thesis the concept of thread features is generalised based on their findings in Chapter 5.

[77] Gkotsis et al. (2014)

In addition to analysing the relation between contribution time (i.e. the elapsed time between a question is posted and an answer contributed) and quality answers (i.e. answerers that highlight reasonableness, soundness, and dependability for a given question), Chua and Banerjee[78] also studied the impact of time on *best answers* compared to manually annotated answers based on three different quality criteria: reasonableness, soundness, and dependability. The results show that *best answers* tend to take more time than answers that only highlight quality components. This result shows that answers quality improves when given more time and that *best answers* criteria may be stricter that the quality answer criteria.

[78] Chua and Banerjee (2013)

The understanding of the relation between asker *best answer* selection process and community ratings via third party annotations also attracted interest.[79] In their work, Shah and Pomerantz[80] asked annotators from Amazon Mechanical Turk to rate 600 answers from Yahoo Answers based on 13 different criteria[81] and created a regression model for identifying *best answers* using such criteria. They obtained 79.50% accuracy. Their findings show that "novel", "original", and "readable" answers are associated with *best answers*. This result largely confirmed that *best answers* are highly related to manually annotated answers meaning that *best answers* and answer quality are highly related. Following such experiments, the authors decided to create a more standard regression on 5032 answers models generated from 21 different features about askers, answers, questions and answers and obtained 84.52%. They identify the position of an answer within a thread as the most important feature (80.34% accuracy alone).

[79] Shah and Pomerantz (2010)

[80] Shah and Pomerantz (2010) Amazon Mechanical Turk, https://www.mturk.com.

[81] Zhu et al. (2009)

3.2.2 *Quality Answers Identification*

A large fraction of existing work has centred on finding the quality of answers based on external annotations not given by the contributors of the analysed community.

The advantage of such method are multiple: 1) The annotations can be more reliable as they are performed in a controlled environment where annotators are given specific directions; 2) the quality annotations may be cross checked between different annotators allowing for more consistent annotations, and; 3) The annotations may be less subject to community or contributor bias and therefore more

objective. Nevertheless, the main drawbacks are that: 1) External annotators may be less able to determine the quality of answers compared to community annotators as they are less familiar to the topic discussed within a particular community; 2) The amount of available annotation may be lower compared to community annotation,and; 3) The annotations may be less representative to what a given community consider important in quality answers.

Perhaps the research that is the closest to the work presented in this thesis is given by Agichtein et al.[82]. In their paper, the authors proposed to train a classifier for identifying manually annotated quality answers in Yahoo Answers using different quality (e.g. readability measures, grammaticality), user (e.g. user reputation through HITS[83] and Page Rank[84]) and usage (e.g. number of views) features. The authors obtained good prediction results with a reported *AUC* of 0.878.

[82] *Agichtein et al. (2008)*

[83] *Kleinberg (1999a)*

[84] *Page et al. (1999)*

The annotations were performed on three different criteria: 1) well-formedness; 2) readability; 3) utility; 4) interestingness, and; 5) correctness. In general, their result show the ability of classification models to identify quality answers particularly when using n-grams textual features. Such a feature was used as a baseline giving $AUC = 0.805$ and was designed by using all the word n-grams up to length 5 with a corpus frequency higher than 3 as a different predictor.

The most important features besides the n-gram feature is found to be the length of answers as well as the reputation of answerer confirming the result of the user study conducted in this thesis on the importance of user expertise in determining quality answers (Chapter 2).

Although, n-grams are found to be good at identifying quality answers, they lack interpretability. Owning to this, such type of feature is not used in the baseline model developed in this thesis. Another issue is that many of the approaches mentioned above including Agichtein et al.'s work are centred either on one community or are limited to the Yahoo Answers community meaning that existing results may not be similar for other communities. Finally, Agichtein et al.[85]'s work investigated external annotations instead of community annotations. Such setting is different to this thesis work where the focus is on *best answer* identification.

[85] *Agichtein et al. (2008)*

Research by Chua and Banerjee[86] investigated the relation between answering speed and quality answers. In their paper, they manually annotated quality answers and labelled questions depending on their type using the five different criteria proposed by Harper et al.[87]. They found different results depending on the type of question answered. Their study was done on seven different communities: Yahoo Answers, WikiAnswers, Answerbag, Baidu Knows, Tencent, Soso Wenwen and Sina iAsk. For factual questions, they acknowledge that good answers tend to take more time than the fastest answers. This result is interesting as it shows that good quality answers need time even though in the user study presented in the previous chapter it appears that askers expect fast answers (Chapter 2). Such importance of prompt answer has been largely confirmed by previous studies[88].

[86] *Chua and Banerjee (2013)*

[87] *Harper et al. (2010)*

[88] *Kitzie and Shah (2011); Mamykina et al. (2011)*

3.2.3 *Matching Existing Answers to New Questions*

Perhaps one of the earliest research related to *best answer* identification is the finding of candidate answers from a pool of available answers for newly asked questions.[89] Such a type of work is different to both the traditional problem of automatic question answering and *best answer* identification. In automatic question answering, answers are automatically generated for new question using a knowledge base or other information sources. In *best answer* identification tasks, the *best answer* to a question is selected from the already posted question answers.

[89] *Jeon et al. (2006); Berger et al. (2000); Surdeanu et al. (2008); Suryanto et al. (2009)*

In their work, Berger et al.[90] analysed Usenet FAQ documents and customer service call-centre dialogues and designed a system that suggests answers from a large database of answers to newly asked questions. Although slightly different to this thesis problem, the aim of their work was to identify the most suitable answer from a set of available answers making it relevant for the task of *best answer* identification. In order to find suitable answers, the authors applied statistical translation models between question and answer. Their findings show that such type of model is relevant for identifying answer candidates.

[90] *Berger et al. (2000)*

Following on their previous work, Surdeanu et al.[91] investigated the application of an LTR model based on the *Perceptron* algorithm for matching existing answers to questions. In their work, the authors identified quality answers based on *best answers* labels and used NLP features for ranking potential answers to questions. In general, the authors obtained good result with $MRR = 64.65$ from their Yahoo Answers corpus. Their results show that standard IR

[91] *Surdeanu et al. (2008)*

methods such as $BM25$[92] provide great accuracy and confirm the importance of translation features between question and answers.

[92] *Robertson and Walker (1997)*

Jeon et al.[93] also investigated the identification of existing answers to newly asked questions but instead of focusing on textual features, they used non-textual features such as available answer ratings coupled with similarity metrics. In their work, their results show that there is no correlation between quality answers and asker rating but it is suggested that such behaviour may be particular to the Korean inclination to appreciate answers even when they are not accurate. In their work, the authors used KERNEL DENSITY ESTIMATION (KDE) for increasing the correlation of features with quality answers. By using such method they showed that better results can be obtained when identifying quality answers. For instance, the correlation between quality answers and answer length increased by around 2.5 times after KDE was used. These results show that normalisation methods can have a dramatic impact on quality answer identification.

[93] *Jeon et al. (2006)*

Suryanto et al.[94] investigated the integration of answer quality into their method for matching answers to new questions. They used external annotators to identify quality answers as informative, useful, objective, sincere, readable, relevant and correct[95]. Their dataset was based on Yahoo Answers and contained 1000 annotated answers. In order to rank potential answers, they combined relevance score with quality scores generated by the HITS algorithm[96] and previous user reputation scores. In order to compute the HITS measure the authors used the reply exchanges between askers and answerers as a method for propagating manually annotated quality scores and reputation information. In general, their results show

[94] *Suryanto et al. (2009)*

[95] *Suryanto et al. (2009)*

[96] *Kleinberg (1999a)*

that reputation propagation improves the accuracy of their answer ranking method compared to non propagation approaches similar to Jeon et al.'s model.[97]

[97] Jeon et al. (2006)

3.2.4 Measuring Asker Satisfaction

Some works have also explored if question askers are likely to obtain answers to their questions.[98] Although different to *best answer* identification, understanding if a given question is likely to find a correct answer may helps to identify quality or best answers.

[98] Agichtein et al. (2009); Liu et al. (2008)

In their work, Agichtein et al.[99] proposed to use different algorithms such as decision trees, SVM and Naive Bayes for predicting if a question will be receiving a satisfying answer. Their approach used features available at question time as well as information about existing answers and contributors. From their 72 features, they designed some relational features but only focused on question-answer relations (e.g. overlap between question and answers, number of answer, etc.) and did not consider answer-answer features. Their findings showed encouraging results with $F1 = 0.77$ on their Yahoo Answers dataset when using the C4.5 decision tree algorithm. In particular, they observed that question features help for identifying the likelihood of getting a *best answer* but that the relational features that they used do not help much for such a setting.

[99] Agichtein et al. (2009); Liu et al. (2008)

3.2.5 Measuring Question Quality

Besides identifying *best answers*, Agichtein et al.'s[100] paper on

[100] Agichtein et al. (2008)

finding quality content in Q&A communities also investigated the identification of quality questions. In order to do so, they applied a similar technique to the one they used for identifying *best answers*. The results are similar to the *best answer* identification task (*AUC* of 0.761). These results show that methods for identifying *best answers* and quality questions may be exchangeable and that previous work on quality question identification may be used for *best answer* identification.

[101] *Li et al. (2012)*

Li et al.[101] considered that quality questions are associated with attractiveness, number of answers and the amount of time it takes to obtain a *best answer* and attempted to build a model for identifying quality questions. The authors developed a model for propagating user expertise and question quality between questions that share similarities which provided average accuracy. In general, they conclude that their approach based on propagation methods performs better than simpler approaches such as logistic regression.

[102] *Harper et al. (2009)*

Related to question quality, Harper et al.[102] tried to create a model for distinguishing factual questions from conversational questions using textual metrics and user metadata. Other work studied why answers do not obtain answers[103] or studied the problem of question recommendations.[104]

[103] *Yang et al. (2011)*

[104] *Dror et al. (2011); Liu and Agichtein (2008)*

3.3 Qualitative Design Features

The previous chapter identified two main areas of investigation as part of the qualitative design methodology developed in this thesis

RQ 1.2). First, it was found that question complexity and user maturity may be used as a proxy measure of user ability to learn new things and be knowledgeable. Second, the ability to measure the reactivity of answers was found as a way to identify *best answers* in the form of contribution effort modelling. The following sections discuss existing work related to question complexity, user maturity and effort modelling.

In addition to investigating such features, methods related to content quality and readability are also discussed as they can be integrated into identification models of *best answers* and can be associated with *best answers* (Chapter 2).

3.3.1 *Quality and Readability*

Much research has investigated the identification and characterisation of quality content. Long before the raise in popularity of online social platforms and Q&A communities, the need of understanding what makes good content has been investigated with the design of different user studies or the creation of textual metrics.

Quality content has been defined differently depending on the type of text to be assessed. In this section, the focus is mostly on answers and how previous works have defined quality content as well as the design of metrics for assessing such quality automatically.

Content Quality and Accuracy: Beside the research on quality answer and *best answer* identification outlined in the previous sections, some work has directly investigated the measurement of content quality.[105]

[105] *Chai et al. (2009); Katerattanakul and Siau (1999); Lee et al. (2002); Anderka et al. (2011, 2012)*

Another line of research involves AUTOMATIC ESSAY SCORING (AES), an area of work designed for automatically attributing marks to textual content or essays using textual features.[106]

[106] *Burstein et al. (1998); Chodorow and Leacock (2000); Burstein and Wolska (2003)*

Some research involved the identification of quality content in websites[107]. In their work, Katerattanakul and Siau[108] identify four factors of quality: 1) intrinsic quality; 2) contextual quality; 3) representational quality, and; 4) information accessibility. They also asked 64 people to rate different websites using the previous criteria and found that their factors are reliable for rating quality information.

[107] *Katerattanakul and Siau (1999)*

[108] *Katerattanakul and Siau (1999)*

In their paper, Lee et al.[109] developed a methodology for estimating the ability of different methods to estimate the quality of a given information. They surveyed different research in order to identify the factors of quality information and identify the same factor of quality found by Katerattanakul and Siau.[110]

[109] *Lee et al. (2002)*

[110] *Katerattanakul and Siau (1999)*

Chai et al.[111] also surveyed existing research on information quality in USER GENERATED CONTENT (UGC). Their work followed Knight and Burn's survey[112] on online information quality. In their survey, the authors analysed different communities including two Q&A: Naver and Yahoo Answers. Their findings show that such communities have mechanisms for allowing the identification of quality content using user feedback (e.g. user reputation and ratings) meaning that reputation and ratings can be used as indicators of quality.

[111] *Chai et al. (2009)*

[112] *Knight and Burn (2005)*

More recently, some research has explored the identification of quality flaws in content rather than quality indicators.[113] In their research the authors proposed to identify Wikipedia articles that have

[113] *Anderka et al. (2011, 2012)*

quality issues using automatic classification. They trained different classifiers for identifying different flows such as unreferenced articles and orphan documents. Using features similar to previous work,[114] the authors obtained very high accuracy for particular issues such as unreferenced articles. In general, structural flaws could be identified easily whereas conceptual issues are harder to spot (e.g. original research).

[114] Hasan Dalip et al. (2009)

Besides the research on information quality, some research has looked at automatically grading essays[115] where the aim is to provide automatic methods for attributing marks to a written essay.

[115] Burstein et al. (1998); Chodorow and Leacock (2000); Burstein and Wolska (2003)

One approach for such a task is described by Burstein et al.[116] where they used a combination of text analysis and topical analysis and regression analysis for predicting the score of different English tests. Their findings show that the textual similarity between the topic of an essay and a written essay is important for identifying quality essays. In a subsequent work,[117] the authors investigated word repetition as an indicator of low quality essays and found that it cannot be used reliably.

[116] Burstein et al. (1998)

[117] Burstein and Wolska (2003)

Chodorow and Leacock[118] investigated the particular issue of grammatical errors as an indicator of low quality content. They found that gramatical error identification to be relatively good to predict badly written essays with 80% accuracy. The authors used a large corpus of words and PART OF SPEECH (POS) bi-grams containing correct usage of words and mutual information measure for comparing texts in essays with their corpus.

[118] Chodorow and Leacock (2000)

Frické and Fallis[119] investigated directly the accuracy of different websites using reference questions and observing if they are answered correctly using external references. In general, the authors found

[119] Frické and Fallis (2004)

that material that was referencing external sources or referenced by external sources particularly peer reviewed ones was more likely to be associated with accurate answers.

Content Readability: Another area of research that can be associated with content quality is the measurement of content readability. Traditionally, a large amount of research has focused on the design of readability metrics for modelling the complexity of text using text analysis and NLP.[120]

[120] *Gunning (1952); Kincaid et al. (1975); Flesch (1948); Brown and Eskenazi (2005); McLaughlin (1969)*

Perhaps one of the oldest readability metrics was proposed by Flesch[121] for determining the difficulty to read and understand a given textual content. The proposed approach uses the number of syllables per words and number of words per sentence to indicate how complex a given text is. The proposed metric was slightly modified as the Flesch-Kinkaid[122] readability metric by for changing the boundaries of the metric to indicate the number of years of education required for understanding a given text. A similar readability metric called Gunning-Fog-index was developed by Gunning[123] as a mean for representing the easiness of a given text.

[121] *Flesch (1948)*

[122] *Kincaid et al. (1975)*

[123] *Gunning (1952)*

The SMOG grade[124] was also proposed for identifying complex texts. Similarly to other readability measures, such measures mostly rely on the number of syllables and words in a sentence. The LIX metric[125] was also proposed as an alternative measure but used long words as well as punctuation for determining complex paragraphs.

[124] *McLaughlin (1969)*

[125] *McLaughlin (1969)*

Although simplistic by nature, readability measures are relatively easy to compute making them attractive for modelling predictors of answer quality. In this thesis, a few of the methods listed above

are reused as part of *best answer* identification model (Chapter 4) and question complexity model (Chapter 6).

3.3.2 *Expertise, Question Complexity and Maturity*

A large amount of research has investigated the representation of user expertise using automated methods. In particular, much research has proposed to use graph algorithms for propagating expertise between users and posted content[126]. In term of community maturity, less research has directly looked at modelling *the ability of community answerers to learn new things* even though it was observed that knowledge improvement motivates user participation in online Q&A websites (Chapter 2).

[126] *Zhang et al. (2007); Jurczyk and Agichtein (2007a); Campbell et al. (2003); Bian et al. (2009); Serdyukov et al. (2008)*

Measuring the maturity of a community depends on different factors that correlate with user reputation, the ability of users to contribute complex content and to help the improvement of a given community. In this thesis, *the needs of users in improving their knowledge over time as a key component of the maturation process of enquiry communities* is considered as mature user should be able to improve their knowledge with their increasing participation over time. As such, three main areas of research that relate to community maturity measurement and expertise in general are investigated: 1) user's skill building and expertise; 2) content complexity, and; 3) community health.

User Expertise and Skills: The user study conducted in this thesis (Chapter 2) and many other studies of Q&A communities have

shown that user motivation lies in their desire to help others and improve their knowledge on particular topics.[127] These studies highlight the contributors' intent of knowledge improvement, meaning that contributors are expected to create more focused and complex content over time.

[127] Butler (2001); Nam et al. (2009); Mamykina et al. (2011)

Skill building is closely related to user expertise, the ability of users to provide quality content. Expertise has often been analysed using graph algorithms.[128] Some other research has investigated the use of the quality of previous contributions or simple textual features.[129] Recently, some researchers have started investigating expertise evolution.[130] Although research on expertise measurement is related with the maturity of communities, it is more centred on evaluating a user's answering abilities without considering her knowledge skills as a asker. This thesis argues that a good measure of community maturity should take into account both askers' and answerers' knowledge skills.

[128] Zhang et al. (2007); Jurczyk and Agichtein (2007a); Campbell et al. (2003); Bian et al. (2009); Serdyukov et al. (2008)

[129] Chen et al. (2011); Dom and Paranjpe (2008)

[130] Pal et al. (2012)

For instance, in their research, Zhang et al.[131] used two popular graph based propagation algorithms: Page Rank and HITS[132] in order to identify experts in Q&A communities. They also used the z-scores for identifying users that are more focused on asking or answering questions and extended Page Rank[133] for expertise propagation. In general they found that z-scores are very good at identifying experts despite being easy to compute. Jurczyk and Agichtein[134] also used the HITS algorithm but extended it for taking into account the topic of a given question and obtained encouraging results.

[131] Zhang et al. (2007)

[132] Kleinberg (1999b)

[133] Page et al. (1999)

[134] Jurczyk and Agichtein (2007a,b)

Similarly, Campbell et al.[135] used the HITS algorithm for a similar

[135] Campbell et al. (2003)

purpose on email discussions and found that experts are associated with high HITS score.

Bian et al.[136] proposed to use mutual reinforcement for learning the expertise of users as well as the quality of questions and answers. They used different features including the hub and authority scores returned by the HITS algorithm[137]. They obtained accurate results for matching answers to query questions and showed better results compared to HITS alone.

[136] *Bian et al. (2009)*

[137] *Kleinberg (1999b)*

Serdyukov et al.[138] introduced a model for finding experts using a textual query. They used random walks for propagating expertises between users. Their approach was evaluated on discussion between users on the W3C website as well as CSIRO data from TREC 2007.

[138] *Serdyukov et al. (2008)*

Instead of using a graph based approach, Chen et al.[139] used ratings on comments for identifying experts in Yahoo Buzz coupled with different textual features such as the length of the comment and lexical diversity. They introduced a latent factor model for identifying quality of comments and showed strong results despite the computational complexity of their model.

[139] *Chen et al. (2011)*

A somewhat simpler approach was proposed by Dom and Paranjpe[140] using a Bayesian model for predicting the credibility of question answerers by predicting the probability of an answerer to give the *best answer*. Their approach relies on the history of community user contributions and reputation.

[140] *Dom and Paranjpe (2008)*

Pal and Konstan[141] used different models of expertise based on the previous number of *best answers* contributed by answerers as well as a measure of question existing value that relies on the current rating of the answers to a given question and other simple metrics.

[141] *Pal et al. (2012)*; *Pal and Konstan (2010)*

They showed that their simple metrics helped to identify expert users and that experts tend to answer questions with low exiting value. A trend that increases with their age in the community.

Content Complexity: Measuring the complexity of content is difficult, therefore, much work has focused on very specific domains where the difficulty of content is relatively easy to measure, e.g., multiple-choice Q&A. In the context of multiple-choice Q&A, the ITEM RESPONSE THEORY (IRT), a paradigm designed for scoring tests and questionnaires, is often used for simultaneously identifying skills and question complexity based on answers. Probabilistic models have also been proposed based on IRT[142] to automatically grade tests. Nevertheless, these approaches cannot be generalised easily to other domains owing to their reliance on particular answers structures.

[142] *Welinder et al. (2010); Bachrach et al. (2012)*

Welinder et al.[143] proposed a probabilistic model for estimating the difficulty of questions as well as user ability and the identification of the correct answer in multi choice Q&A. Their approach relies on Bayesian modelling with minimal input however, the proposed DARE model only work on predefined answers and cannot be applied directly to the type of Q&A communities analysed in this thesis. Bachrach et al.[144] also used a Bayesian model but instead of identifying correct answers, they designed it for finding images that are difficult to annotate making it not directly applicable to the task of identifying complex questions.

[143] *Welinder et al. (2010)*

[144] *Bachrach et al. (2012)*

Some work considered complex questions as questions that require longer answers such as non-factual questions or definition questions[145]. Although such questions are more likely to be harder than

[145] *Lin and Demner-Fushman (2006); Lin and Zhang (2007)*

simple questions, their definition differs from our more general definition of complex questions.

Community Health: Community maturity can be seen as an extension of existing community health metrics.[146] Wu[147] defined a COMMUNITY HEALTH INDEX (CHI) measure that uses thread closures, content popularity and web traffic for measuring the performance of enquiry communities.

[146] *Toral et al. (2009); Sterne (2010); Rowe and Alani (2012)*

[147] *Wu (2009)*

Based on the prior research on community health analysis, Rowe and Alani[148] identified four different health factors: loyalty, participation, activity and social capital. Surprisingly, few works considered the ability of a community to generate knowledgeable users as a health metric despite the fact that gaining knowledge is often considered as one of the most important reasons for users to participate in a community.[149]

[148] *Rowe and Alani (2012)*

[149] *Nam et al. (2009); Mamykina et al. (2011)*

In this thesis, maturity is defined as representing an important process in community evolution. The main idea is that a mature community is a community that generates more knowledgeable content and hence is able to fulfil user needs more easily. In this aspect, maturity differs from existing community health measures at user contribution and expertise levels since it takes into account the evolution of of Q&A communities: the bilateral (i.e. askers and answerer) knowledge build-up process involved in community consolidation. The method used for creating a measure of community maturity is introduced in Chapter 6 as part of the qualitative design methodology of this thesis.

3.3.3 Community reactivity and Contribution Effort

In this thesis, contribution effort is modelled instead of only relying on time-to-answers features for modelling the time component of *best answers* as identified in Chapter 2. Contribution effort can be defined as *the amount of work required for a given user to produce a particular answer*.

Existing research on effort has mostly focused on topics indirectly related to effort such as user expertise[150] and community attention.[151] Finally, topic modelling can be linked to the problem of effort modelling as the amount of work required for posting may be connected to topic dynamics and vocabulary usage for a given user.[152] This relation is further explained and studied in detail in Chapter 7 where the model of contribution effort is designed.

User Expertise and Skills: User expertise can be linked to effort modelling and community reactivity as it can be assumed that knowledgeable users may be more likely to contribute faster than less knowledgeable users when answering their favourite topic. To some extent, such a relation has been studied by Chua and Banerjee[153] when they analysed the relation between quality answers and answering velocity and found that good answers need more time for being submitted (Section 3.2.2). Nevertheless, expertise is not enough to measure contribution effort as it can be expected that answering time may also be linked to simpler questions or short answers. In general the observations about expertise modelling are the same as discussed in Section . Expertise modelling generally either use graph algorithms[154] or community feedback.[155] In any case, these

[150] Zhang et al. (2007); Jurczyk and Agichtein (2007a); Agichtein et al. (2008)

[151] Rowe et al. (2011b); Wagner et al. (2012b,a); Rowe and Alani (2014); Mathioudakis et al. (2010); Ruiz et al. (2014)

[152] Blei et al. (2003); Lin and He (2009); Rosen-Zvi et al. (2004); Blei and Lafferty (2006); He et al. (2014); Chang and Blei (2009); Liu et al. (2009)

[153] Chua and Banerjee (2013)

[154] Zhang et al. (2007); Jurczyk and Agichtein (2007a); Campbell et al. (2003); Bian et al. (2009); Serdyukov et al. (2008)

[155] Chen et al. (2011); Dom and Paranjpe (2008)

approaches do not properly model the contribution effort of users as well as the reactivity of community members.

User Attention: A topic closely related to contribution effort is attention modelling where the goal is to identify the questions or posts that are most likely to attract contributions. This type of research is closely related with question quality and asker satisfaction as quality questions are more likely to attract more answers (Subsections 3.2.4 and 3.2.5).

Work on user attention has mostly focused on the collective attention of users rather than individual ones by identifying questions that generate the most answers.[156] Similarly, other research has proposed to analyse the behaviour of users in order to better understand the content that generates high activity by taking in account time dynamics.[157]

In their research, Rowe et al.[158] constructed different models for identifying posts that generate comments or seed posts in online forums and a model for predicting the amount of comments generated by such posts. Their approach relied on three type of features: 1) user features; 2) content features, and; 3) focus features. The focus features contained the topical concentration of users across topics as well as the likelihood of users to contribute to a particular topic. They managed to predict accurately seed posts ($F_1 = 0.792$). They observed that the topical focus associated with seed post is more likely to generate longer discussion or attention from a community.

Following on the previous research, Wagner et al.[159] focused their attention on sub communities of a forum community and found that

[156] *Rowe et al. (2011b); Wagner et al. (2012b,a); Rowe and Alani (2014)*

[157] *Mathioudakis et al. (2010); Ruiz et al. (2014)*

[158] *Rowe et al. (2011b)*

[159] *Wagner et al. (2012b,a)*

the features that generate attention is largely dependent on the topic discussed.

[160] *Rowe and Alani (2014)*

In a subsequent work, Rowe and Alani[160] extended their research to more communities including the communities studied in this thesis and confirmed that attention dynamics depend on the community under investigation. They found that for both the SCN and SF communities, readability measures and informativeness metrics were associated with seed posts.

[161] *Mathioudakis et al. (2010)*

Mathioudakis et al.[161] used a stochastic model for identifying content that gather attention in blogs. The authors used a sequential model for taking into account temporal dynamics that affect user attention.

The relation between efficiency, the amount of attention received in relation to the content produced by a user, and attention was investigated by Ruiz et al..[162] The author analysed Yahoo! Meme and found that successful content generally depends on post authors to maintain attention to their content over time by keeping conversations alive through additional activity.

[162] *Ruiz et al. (2014)*

In general, the above works only acknowledge the community wide attention that posts gather rather than the individual attention associated with a given user. Although contribution effort depends on the user interest to contribute and therefore her attention, attention does not take into account the hidden cost of contributions: *the amount of work that created a post*.

Topic Modelling and Effort Modelling: As previously observed, existing research has mostly focused on expertise models and attention modelling instead of focusing on effort modelling: *the representation of the hidden amount of work required by a user to contribute*.

Bayesian modelling is particularly suitable for inferring latent information such as contribution effort from observed data as it allows for the definition of structural relation between observed and unobserved variables.[163]

[163] *Paquet (2007)*

Topic models (Chapter 2) such as LDA[164] explain the words observed in documents from latent topics. Topic models have the advantage of automatically deriving topic themes from documents without supervision. Therefore, topic models seem appropriate for modelling effort given their unsupervised nature. As observed in the previous chapter, much research has proposed to extend topic models model with additional latent variables in order to do add extra levels of predictions.[165]

[164] *Blei et al. (2003)*

[165] *Lin and He (2009)*; *Rosen-Zvi et al. (2004)*

For instance, Lin and He[166] proposed the JST model to represent topic and topic-associated sentiment by using a word-sentiment prior. Another work by Rosen-Zvi et al.[167] introduced the Author Topic (AT) model for modelling authors and author-specific topics.

[166] *Lin and He (2009)*

[167] *Rosen-Zvi et al. (2004)*

Further extensions considered the evolution of topics over time[168] as well as sentiment.[169] For example, Blei and Lafferty[170] proposed the Dynamic Topic Model (DTM) and included time information for representing time dynamics in topics whereas He et al.[171] proposed the DYNAMIC JOINT SENTIMENT TOPIC MODEL (DJST) model for modelling the evolution of sentiments in topics by integrating time dependencies on topics.

[168] *Blei and Lafferty (2006)*

[169] *He et al. (2014)*

[170] *Blei and Lafferty (2006)*

[171] *He et al. (2014)*

Since the contribution of effort of users is likely to change over time, the integration of time features similarly to the previous approach may benefit a topic model of contribution effort.

Another approach for identifying temporal relations between topics may be obtained by connecting a time-independent topic model with a time-dependent topic model. Linked topic models has been used for modelling relational information.[172] However, linking time dependent models with time independent models for tracking the evolution of a given variable over time has not been studied before.

[172] *Chang and Blei (2009); Liu et al. (2009)*

For example, Chang and Blei[173] proposed a model for identifying the connection between different topics using connections between cited documents and citing documents. Another model that investigated topic relations was proposed by Liu et al..[174] The authors designed a model for connecting document authors and topic communities.

[173] *Chang and Blei (2009)*

[174] *Liu et al. (2009)*

Even though existing models have not addressed the issue of modelling contribution effort, a similar approach may be used for identifying user specific contribution effort for particular topics. In chapter 7, a topic model based on the JST[175] model is proposed for modelling contribution effort, one of the features identified in the previous chapter as part of the qualitative design approach studied in this thesis (RW 1.2).

[175] *Lin and He (2009)*

3.4 Discussion

The previous sections highlighted existing research related to *best answer* identification and qualitative design features. The next section discuss the limitation of such works and presents the main differences between previous research and the work conducted in this thesis.

3.4.1 General Observations

The main difference and novelty between this thesis methodology and previous work is the application of structural and qualitative analysis for improving *best answer* identification models. Both of these issues have not really been studied previously and even less together. In particular, previous research as either focused on qualitative studies independently from feature design or on the improvement of identification models using graphical metrics instead of structural optimisation methods.

The graph metrics used in the works listed in the previous section (e.g. Page Rank[176] and HITS[177]) are quite different from structural optimisation techniques developed in this thesis: 1) graph metrics focus on only individual predictor instead of the whole optimisation of existing metrics (i.e. the systematic structural normalisation of *best answer* predictors); 2) existing approaches are focused on reputation propagation instead of the various other aspects that make an answer a *best answer*, and; 3) the existing methods tend to not take into account the relations between the answers of a given thread.

[176] Page et al. (1999)

[177] Kleinberg (1999b)

As highlighted in the previous section, only few work actually take into account the structure of Q&A communities.[178] However, these approaches are limited in scope and are non-systematic compared to the methods introduced in Chapter 5.

[178] Agichtein et al. (2009); Liu et al. (2008); Tian et al. (2013); Gkotsis et al. (2014)

Another important difference between existing work and the proposed research is that previous approaches tend to only focus on the raw improvement of identification accuracy whereas this thesis is more interested on the impact of structural and qualitative methods for improving classification algorithms in the context of *best answer* identification.

Even though the ultimate goal of this work is to enable the improvement of *best answer* identification, the main aim is to determine if systematic methods can used for improving *best answer* identification in general. The advantage of such aim is that the results can help the future development of additional features and algorithm using a sound and clear methodology.

3.4.2 Best Answer Identification

Very few research works investigated the application of specific models that take into account the thread-like structure of Q&A communities with the most successful approaches relying on decision tree algorithms.[179] Since decision tree models seem to perform very well for identifying *best answers*, the *best answer* models developed in this thesis are all based on decision tree models. In particular alternating decision tree algorithms are used extensively in this body of work due to their ability to perform good predictions (Chapter 4).

[179] Agichtein et al. (2009); Liu et al. (2008); Tian et al. (2013); Gkotsis et al. (2014)

The usage of more specific algorithms has been applied to related tasks such as answer matching. In particular, LTR models have proven their usefulness for associating existing answers with new questions by relying on their ability to rank answers automatically.[180] Although LTR models have not been used for automatically identifying *best answers*, the ability of such models to do ranking can be used for ranking answers. In this thesis, the usage of LTR models is investigated in Chapter 5 as part of the structural design methodology presented in this thesis (RQ 1.1).

[180] *Surdeanu et al. (2008)*

In the area of feature normalisation, existing works tend to not use any specific normalisation techniques. In the previously mentioned existing work, Jeon et al.[181] used KDE for improving the association of existing quality answers to new questions. Although the proposed approach shows improvement, the approach does not take into account the structure of Q&A communities and is not a type of structural optimisation. Moreover, the method was not used in the context of *best answer* identification. The only existing approach that performs some sort of structural optimisation based on the thread structure of Q&A communities was proposed by Gkotsis et al..[182] In their work, the authors used the rankings of different features within a thread instead of their actual value improving upon some of the thread features that are introduced in Chapter 4. In general, such results show the usefulness of such a method. This thesis considers and investigates a generalised approach to such a method in Chapter 5 as part of the structural design (RQ 1) approach developed in the thesis.

[181] *Jeon et al. (2006)*

[182] *Gkotsis et al. (2014)*

The integration of qualitative design (RQ 1.1) on the design of *best answer* model has been generally lacking as previous feature

designs have mostly been based on intuition. Although some features that can be related to contribution effort and community maturity have been used in different models such as user reputation[183] and answering velocity,[184] such approaches do not model directly the concepts investigated in this thesis as part of the qualitative design methodology (RQ 1.1). As a consequence, new models are necessary for measuring question complexity, user maturity and contribution effort. Such approaches are introduced in Chapter 6 and Chapter 7.

[183] Blooma et al. (2008); Gkotsis et al. (2014); Agichtein et al. (2008)

[184] Tian et al. (2013); Gkotsis et al. (2014)

3.4.3 Qualitative Design Features

Section 3.3 identified some metrics that can be used for measuring the clarity of content and showed that content quality is an intrinsic part of *best answer* identification and therefore it is not necessarily to create separate quality metric in the context of *best answer* identification. Based on this observation, this thesis feature development focus on other metrics such as user maturity and contribution effort. Despite this focus, a few measures associated with content quality and clarity such as the Gunning-Fog index, the Flesch-Kincaid Grade and the user ratings are used as part of the feature rich *best answer* identification model created in Chapter 4 as well as in Chapter 6. These features are part of the content features used in this thesis.

Works related to user maturity are mostly focused on expertise modelling and generally focus on the ability of users to answer and their *reputation*. In particular, some of such models are based on *reputation* propagation though the application of graph algorithms such

as Page Rank[185] and HITS[186]. From the literature, it can be observed that very little research has been done for differentiating the users that are able to answer complex questions compared to easy questions. Similarly, modelling the progression of users towards acquiring knowledge (i.e. the ability to learn new things) is largely left out from previous works. A novel approach for modelling question complexity and community maturity is therefore introduced in Chapter 6.

[185] Page et al. (1999)
[186] Kleinberg (1999b)

In the domain of identifying the effort of user contributions, existing work has mostly centred on representing user attention and expertise rather than the particular issue of contribution effort. Notable research either involved attention and reactivity modelling or information response theory. In order to provide a model that includes the implicit and unaccounted effort of user contribution, this thesis proposes to use Bayesian modelling (i.e. topic models) to model such a problem. The approach is presented in details in Chapter 7.

3.5 SUMMARY

In this chapter existing work related to *best answer* identification was discussed as well as existing methods for modelling features related to question complexity, community maturity and contribution effort.

The analysis shows that existing works have mostly analysed single communities without focusing much on the determination of the

factors that influence *best answers*. It also shows that although different work has been conducted for modelling factors that influence the identification of quality answers, research has generally ignored the results of qualitative studies to motivate the inclusion and design of particular predictors such as the ones investigated in this thesis.

The study of existing *best answer* identification models and related approaches has been mostly based on standard classification models with decision tree based approaches which have proven to be generally very successful. Structural optimisation methods have been largely left out with the exception of few promising works considering thread-wise normalisation[187] and non specific feature normalisation.[188] These encouraging results prompt the generalisation of thread-wise normalisation approach as well as more specific *best answer* identification algorithms (Chapter 5).

[187] Gkotsis et al. (2014)

[188] Jeon et al. (2006)

In order to create more accurate *best answer* classifiers, this thesis proposes to produce two new quality predictors based on the literature and the user study presented in the previous chapter (Chapter 2): 1) a measure of content complexity (Chapter 6), and; 2) A measure of contribution effort (Chapter 7). Besides being based on the qualitative study findings from different user studies (Chapter 2), each measure departs from the literature by respectively measuring the ability of answerers to answer hard questions and to estimate the hidden amount of work going into of user contributions.

In the following chapter, an initial model designed for identifying *best answers* is presented. This model is used as a baseline for identifying *best answers* in Q&A communities. The proposed model is fitted with a large number of features obtained from the

presented literature and introduces some new features. Following this preliminary model, different structural optimisation approaches are proposed and evaluated (Chapter 5). The evaluation of the question complexity, maturity and contribution effort models are evaluated in the third part of this thesis as part of the study of the qualitative design methodology presented in this thesis (RQ 1).

Part II

STRUCTURAL DESIGN AND BEST ANSWER IDENTIFICATION

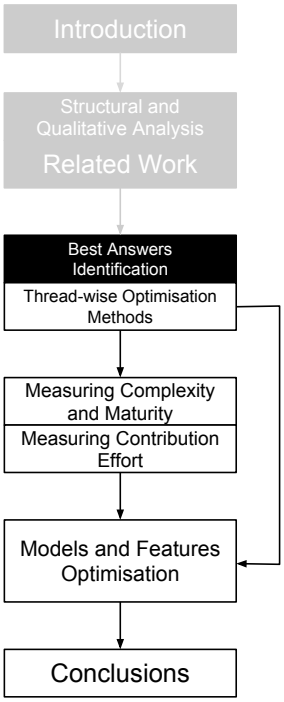
4

BEST ANSWER IDENTIFICATION

The evaluation and study of methods for improving the automatic identification of *best answers* requires the development of a model that can be used as the base for testing the structural and qualitative design methodologies studied in this thesis (RQ 1).

In order to automatically identify *best answers*, the availability of *best answer* labels in the datasets described in chapter 2 are used for training binary classifiers using different *users*, *content* and *thread* features. Such features follow the feature classification approach presented in chapter 3.

By using such a type of classifier, the proposed classifiers are able to provide encouraging prediction results that can be used as the basis for the enhanced models created in the following chapters that integrate structural optimisation methods (Chapter 5) and features designed through qualitative design (Chapter 8). In particular the few thread features used in this chapter show high association with *best answers* meaning that thread-wise optimisation can be used effectively for improving *best answer* identification models.



This chapter is divided in four different parts. First, the context determining the applications of *best answers* is discussed. Second, different features used for predicting quality answers are introduced. Finally, the *best answer* model is presented before the results are discussed and the chapter summarised.

4.1 INTRODUCTION

It is very common for popular Q&A websites to generate many replies for each posted question. For example, in the datasets studied in this thesis, on average each question thread received 9 replies, with some questions attracting more than 100 replies. With such a mass of content, it becomes vital for online community platforms to put in place efficient policies and procedures to allow the discovery of *best answer*. This allows community members to quickly find prime answers, and to reward those who provide quality content.

As discussed in chapter 2, the process adopted by Q&A systems for rating *best answer* range from restricting answer ratings to the author of the question (e.g. the SCN[189] forums), to opening it up to all community members (e.g. SE). What is common between most of such communities is that the process of marking *best answer* is almost entirely manual. The side effect is that many threads are left without any such markings. In the studied datasets, about 50% of the threads lack pointers to the *best answer*. Although the lack of this label may mean that a question has no solution in many cases it can be expected that question authors simply forget to acknowledge a *best answer*.

[189] *SAP Community Network*, http://scn.sap.com.

In this chapter different models for identifying *best answer* on the three Q&A communities (Chapter 2) are created: 1) The SAP COMMUNITY NETWORK (SCN) forums; SERVER FAULT (SF)[190], and; 3) The COOKING (CO) community[191]. These models form the basis of the models used for evaluating the structural and qualitative design methodology studied in this thesis (RQ 1).

[190] *Server Fault*, http://serverfault.com.

[191] *Cooking community*, http://cooking.stackexchange.com.

The models are tested using various combinations of *user*, *content*, and *thread* features to discover how such groups of features influence *best answer* identification. This chapter also study the impact of community-specific features to evaluate how platform design impacts *best answer* identification. Accordingly, the main contributions of this chapter are:

1. Perform a comparative study on performance of a model for *best answer* identification on three online enquiry communities.

2. Introduce a new set of features based on the characteristics and structure of Q&A threads that is used through this thesis.

3. Study the influence of user, content, and thread features on *best answer* identification and show how combining these features increases accuracy of *best answer* identification.

4. Introduce several ratio features, e.g. *ratio of scores of an answer in comparison to others*; and show that such ratio features have a good impact on our models. These features are generalised in the following chapter (Chapter 5) while designing structural features based on the thread-like structure of Q&A communities.

5. Investigate the impact of platform-specific features on performance of *best answer* identification, and demonstrate the value of public ratings for *best answer* prediction.

6. Investigate whether *best answer* can be still predicted when no thread specific rating are available (e.g. when questions and answers are too recent).

4.2 PREDICTING BEST ANSWERS

Identifying *best answer* requires the training and validation of prediction models and discovering the influence of the various features on these predictions. For training the answer classifier that forms the basis of the *best answer* identification model used in this thesis, three main types of features are used; *content*, *user*, and *thread* features. All these features are strictly generated from the information available at the time of feature extraction (i.e. future information is not taken into account while generating user attributes) and are based on the features types discussed in chapter 3.

In this thesis, these features are applied to different contexts such as when prediction models are optimised (Chapter 5) and when the complexity of questions is measured (Chapter 6). Depending on the context, slightly different feature type classifications are used. The idea of grouping features based on *users*, answering *threads* and *content* characteristics (e.g. individual questions or answers) is kept across each experiment and based on the observations done in Section 3.2.1.

A majority of the features listed below are taken from the related work (Chapter 3) and from features commonly used in document classification tasks as the goal of this chapter is to create a baseline *best answer* classification model that can be tested against the qualitative and structural optimisation techniques proposed in this thesis. The novel features are based mostly on intuition (e.g. *normalised topic entropy* and *topic reputation*) and the small amount of relational features come from the structural analysis performed in the Chapter 2 and are designed for guiding the development of the thread-wise optimisation methods discussed in Chapter 5.

4.2.1 User Features

User features describe the characteristics and reputation of authors of questions and answers. Below is the list of 18 user features employed in this chapter.

- *Reputation*: Represents how active and knowledgeable a user is. It can be approximated from the number of good answers written by the user and the received votes.
- *Community Age*: The user age in the community in days since her first contribution.
- *Post Rate*: Average number of questions or answers the user posts per day.
- *Asking Rate*: Average number of questions the user posts per day.
- *Answering Rate*: Average number of answers the user posts per day.

- *Number of Answers*: The number of answers posted by a user.
- *Number of Posts*: The number of answers and questions posted by a user.
- *Answers Ratio*: The proportion of answers posted by a user compared to her total number of posts.
- *Number of Best Answers*: The number of *best answer* posted by a user.
- *Best Answers Ratio*: The proportion of *best answer* posted by a user compared to her number of posted answers..
- *Number of Questions*: The number of questions posted by a user.
- *Questions Ratio*: The proportion of questions posted by a user compared to her total number of posts.
- *Number of Solved Questions*: The number of questions posted by a user that received a solution.
- *Solved Questions Ratio*: The proportion of questions posted by a user that received a solution.
- *Z-score:*[192] Given a user u_i, calculates her inclination $Z(u_i)$ to ask or answer questions given her question set Q_{u_i} and answer set A_{u_i}. If the user asks more questions than she answers, $Z(u_i) < 0$. Otherwise, $Z(u_i) > 0$. The formula is simpler to

[192] Zhang et al. (2007)

the standard z-score formula as the distribution contains only two set of values (i.e. Q_{u_i} and A_{u_i}):

$$Z(u_i) = \frac{|A_{u_i}| - \frac{|A_{u_i}|+|Q_{u_i}|}{2}}{\frac{\sqrt{|A_{u_i}|+|Q_{u_i}|}}{2}} \tag{6}$$

$$= \frac{|A_{u_i}| - |Q_{u_i}|}{\sqrt{|A_{u_i}| + |Q_{u_i}|}} \tag{7}$$

– *Normalised Activity Entropy*: A normalised entropy measure (H_a) represents how predictable the activity of a user is. In enquiry platforms, a user u_i can either post questions (Q) or answers (A). Lower entropy indicates focus on one activity. The normalised activity entropy is calculated from the probabilities of a user posting answers or questions:

$$H_A(u_i) = -\frac{1}{2}\left(P(Q|u_i)\log P(Q|u_i) + P(A|u_i)\log P(A|u_i)\right) \tag{8}$$

– *Normalised Topic Entropy*: Calculates the concentration (H_T) of a user's posts across different topics. Low entropy indicates focus on particular topics. In our case, topics are given by the tags associated with a question or the category of the post. Each user's tags T_{u_i} are derived from the topics attached to the questions asked or answered by the user. This can be used to calculate the probability $P(t_j|u_i)$ of having a topic t_j given a user u_i:

$$H_T(u_i) = -\frac{1}{|T_{u_i}|}\sum_{j=1}^{|T_{u_i}|} P(t_j|u_i)\log P(t_j|u_i) \tag{9}$$

– *Topical Reputation*: A measure of the user's *reputation* with a particular post. It is derived from the topics T_{q_k} associated

with the question q_k for which the post belongs. By adding the score values of each user's answers $S(a)$, where $a \in A_{u_i,t_j}$, about a particular topic t_j, the general user topical reputation $E_{u_i}(t_j)$ is obtained for a particular topic. Given a post user u_i, the user topical reputation function E_{u_i} and a question q with a set of topics T_q, the reputation embedded within a post related to question q is given by:

$$E_P(q,u_i) = \sum_{j=1}^{|T_q|} \frac{E_{u_i}(t_j)}{\sum_{a \in A_{u_i}} S(a)} \tag{10}$$

$$E_{u_i}(t_j) = \sum_{a \in A_{u_i,t_j}} S(a) \tag{11}$$

4.2.2 Content Features

Content features represent the attributes of questions and answers, and are used for estimating the quality of a particular question or answer as well as their importance. We use the following content features in our analysis:

- *Score:* Represents the rating of an answer. It is collected from users in the form of *votes* or *thumbs up/thumbs down* flags.
- *Answer Age:* Difference between the question creation date and the date of the answer.
- *Number of Question Views:* The number of views or hits on a question.
- *Number of Comments:* The number of comments associated with an answer.

– *Number of Words:* The number of words contained in a question or answer.

– *Readability with Gunning Fog Index*: Used to measure post (i.e. question or answer) readability using the Gunning index of a post p_i which is calculated using the average sentence length asl_{p_i} and the percentage of complex words pcw_{p_i}:

$$G_{p_i}(asl_{p_i}, pcw_{p_i}) = 0.4\ (asl_{p_i} + pcw_{p_i}) \tag{12}$$

– *Readability with Flesch-Kincaid Grade*: Calculated from the average number of words per sentence $awps_{p_i}$ and average number of syllables per word $aspw_{p_i}$ of a post p_i:

$$FK_{p_i}(awps_{p_i}, aspw_{p_i}) = 0.39\ awps_{p_i} + 11.8\ aspw_{p_i} - 15.59 \tag{13}$$

– *Cumulative Term Entropy*: Represents the distribution of words within a question or answer using cumulative entropy. A document containing a variety of different words is potentially more complex to understand. Given a question q_i, its total number of words $|T_{q_i}|$ and the frequency of each word $|t_{q_{i_j}}|$, the cumulative term entropy $C_d(q_i)$ of a question is defined as:

$$C_d(q_i) = \sum_{j=1}^{|T_{q_i}|} \frac{|t_{q_{i_j}}| \times (\log |T_{q_i}| - \log |t_{q_{i_j}}|)}{|T_{q_i}|} \tag{14}$$

4.2.3 Thread Features

The final set of features represents relations between answers in a particular thread. Relational features such as the proportion of votes to a particular answer can be used for estimating the relative importance of a particular post. Theses features can be considered as structural features as they take into account the structure of Q&A communities. They are explored and generalised in the next chapter (Chapter 5).

- *Score Ratio*: The proportion of scores given to an answer from all the scores received in a question thread.
- *Number of Answers*: Number of answers received by a particular question.
- *Answer position*: The absolute order location of a given answer within a question thread according to the posting time (e.g. first, second). This feature represents the depth of an answer in a given thread.
- *Relative Answer Position*: The relative position of an answer within a question thread. Given a question q, its answers a_q, and the position of an answer $pos_{a_{q_i}}$, the relative answer position of an answer a_{q_i} is given according to its chronological order by:

$$RP(a_{q_i}) = 1 - \frac{pos_{a_{q_i}}}{|a_q|} \tag{15}$$

- *Topical Reputation Ratio*: The proportion of topical reputation associated with a particular answer. Given the sum of topical reputation of all the answers, the ratio of topical reputation attributed to a particular answer.

4.2.4 *Core vs Extended Feature Sets*

As mentioned, the impact of platform-specific features on the predictability of *best answer* is investigated. Hence the features above contain some that are not common across the datasets. For example, in SCN only the owner of a question can rate its answers, and select the *best answer*, whereas in the SF and CO communities anyone with over 200 points of reputation can vote for any answer, and hence the selections of *best answer* can emerge collectively. The platform that supports SF and CO offer more features than SCN. Table 9 lists the *core features set*, which is shared across all three datasets, and the *extended features set*, which is only valid for SF and CO datasets.

	Features Set	
Type	Core Features Set (28)	Extended Features Set† (31)
User	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)
Content	*Answer Age, Number of Question Views, Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (6)	***Score**, Answer Age, Number of Question Views, **Number of Comments**, Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (8)
Thread	*Number of Answers, Answer Position, Relative Answer Position, Topical Reputation Ratio.* (4)	***Score Ratio**, Number of Answers, Answer Position, Relative Answer Position, Topical Reputation Ratio.* (5)

†Only valid for the *Server Fault* and *Cooking* datasets.

Table 9: Differences between the Core Features Set and the Extended Features Set. The features in **bold** highlight the differences between the Core and Extended sets.

4.2.5 Stable and Evolving Features

Some features available for identifying *best answers* require some time between the moment a question has been asked or an answer posted and the time a particular feature is realised. For example, community ratings of questions and answers are contributed incrementally over time by individual community users whereas features like the length of a given post does not really varies after submission time.

Althougt, it is technically posibble to change post content over time, the dataset analysed in this thesis do not contain such information. Therefore, it is considered that content length is a stable feature.

In this context, *best answer* identification models that use such dynamic and evolving features can only perform *best answer* prediction when these features do not continue to change anymore (e.g. when users stop rating a given post). In order to determine if the omission of such evolving features impact *best answer* identification, the features mentioned above are split between stable features, which do not change over the course of the evolution of a particular answering thread; and evolving features, features that change after particular contributions. The difference between the extended feature set described in the previous section and the stable features set are listed in Table 10.

It is important to note that the stable feature set includes all the *user features* as they are computed at the moment a user contributes. Therefore, these metrics do not change after a particular user contribution event though users features will change with their future activities.

	Features Set	
Type	Extended Features Set (31)	Stable Features (24)
User	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)
Content	***Number of Comments***, *Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (8)	*Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (4)
Thread	***Score Ratio***, ***Number of Answers***, *Answer Position*, ***Relative Answer Position***, *Topical Reputation Ratio.* (5)	*Answer Position, Topical Reputation Ratio.* (2)

Table 10: Differences between the Extended Features Set and Stable Features Set. Features in **bold** represent dynamic features.

4.3 BEST ANSWER IDENTIFICATION

Ability to accurately identify *best answer* automatically is not only a compliment to the fitness and precision of the prediction model, but also to the fit of the community and platform features that are enabling such task to be performed accurately. If a platform fails to support the gathering of information that correlates with content quality, then automating content quality prediction becomes much harder. More importantly, such difficulty is also be faced by the users who need to quickly find the *best answer* to their problems.

The experiment described next aims at measuring the importance of the core and extended feature sets for *best answer* prediction, as well as highlighting how each feature impacts prediction accuracy in a given platform.

4.3.1 Experimental Setting

In these experiments a categorical learning model is trained for identifying the *best answer* in the three datasets studied in this thesis (Chapter 2). For each thread, the *best answer* annotation is used for training and validating the model. Because the SCN forum *best answer* annotation is based on the author ratings, the *best answer* rating (i.e. 10) is used as the model class and the other ratings are discarded (i.e. 2 and 6) for training the SCN model.

As discussed in Chapter 2, a standard 10-folds stratified cross validation scheme and the *Alternating Decision Tree* learning algorithm is applied for evaluating the generated model. Each model uses the features described earlier so that each training and evaluating instance contains the *user* and *content features* of the related question and answer to evaluate as well as the associated *thread features*.

To evaluate the performance of the learning algorithm, precision (P), recall (R) and the harmonic mean F-measure (F_1) are used as well as the area under the ROC measure. Depending on the target application of the *best answers* prediction models, the result of different evaluation measures may be preferred. For example, if the goal is to identify the best answer when looking at a question, precision may be the best measure. If trying to annotate potential *best answers*, recall can be seen as more important as it is important to miss out any *best answers*. In this thesis, the focus is on the F-measure as it is a evaluation metric that equally accounts for precision and recall.

Two experiments are performed, the first compare the performance of the new model for identifying *best answer* across all three datasets, using the core and extended feature sets as well as the core and extended stable features sets. The second experiment focuses on evaluating the influence of each features on *best answer* identification.

4.3.2 *Results: Model Comparison*

For the first experiment, the *Alternating Decision Tree* classifier is trained on different feature subsets and the results are compared using the metrics that are described in the previous section (Table 11).

Table 11: Average *Precision*, *Recall*, F_1 and *AUC* for the *SCN Forums*, *Server Fault* and *Cooking* datasets for different feature sets and extended features sets (marked with $+$) and reduced features sets (marked with $-$) using the *Alternating Decision Tree* classifier. *All* denotes the combined core *user*, *content* and *threads* features sets. *All+* represents the extended *user*, *content* and *threads* features sets. *All-* is similar to *All* but with only stable features.. *All*$\pm$ is similar to *All+* but with only stable features.

	SCN Forums				Server Fault				Cooking			
Features	*P*	*R*	F_1	*AUC*	*P*	*R*	F_1	*AUC*	*P*	*R*	F_1	*AUC*
Nb. of Words	0.500	0.360	0.419	0.611	0.520	0.588	0.552	0.566	0.565	0.654	0.606	0.655
Answer Score	-	-	-	-	0.592	0.635	0.613	0.673	0.692	0.718	0.704	0.795
Answer Sc. Ratio	-	-	-	-	0.783	0.801	0.792	0.848	0.821	0.842	0.831	0.909
Users	0.572	0.669	0.617	0.754	0.595	0.631	0.613	0.669	0.583	0.651	0.615	0.685
Content	0.551	0.657	0.600	0.674	0.589	0.639	0.613	0.674	0.622	0.686	0.653	0.738
Threads	0.725	0.790	0.756	0.860	0.720	0.745	0.733	0.807	0.655	0.776	0.711	0.780
All	0.752	0.812	0.781	0.883	0.725	0.779	0.751	0.829	0.686	0.765	0.724	0.818
All-	0.710	0.797	0.751	0.850	0.694	0.765	0.728	0.801	0.684	0.741	0.711	0.808
Users+	-	-	-	-	0.595	0.631	0.613	0.669	0.583	0.651	0.615	0.685
Content+	-	-	-	-	0.681	0.691	0.686	0.760	0.732	0.760	0.745	0.842
Threads+	-	-	-	-	0.822	0.840	0.831	0.907	0.820	0.856	0.838	0.914
All+	-	-	-	-	0.823	0.844	0.833	0.911	0.817	0.851	0.834	0.914
All$\pm$	-	-	-	-	0.725	0.779	0.751	0.829	0.686	0.765	0.724	0.818

Baseline Models: The *number of words* feature is used to train a baseline model since it has been argued to be a good predictor. [193] Additionally, for the SF and CO datasets, another basic model based on *answer scores* and *answer scores ratios* is trained since

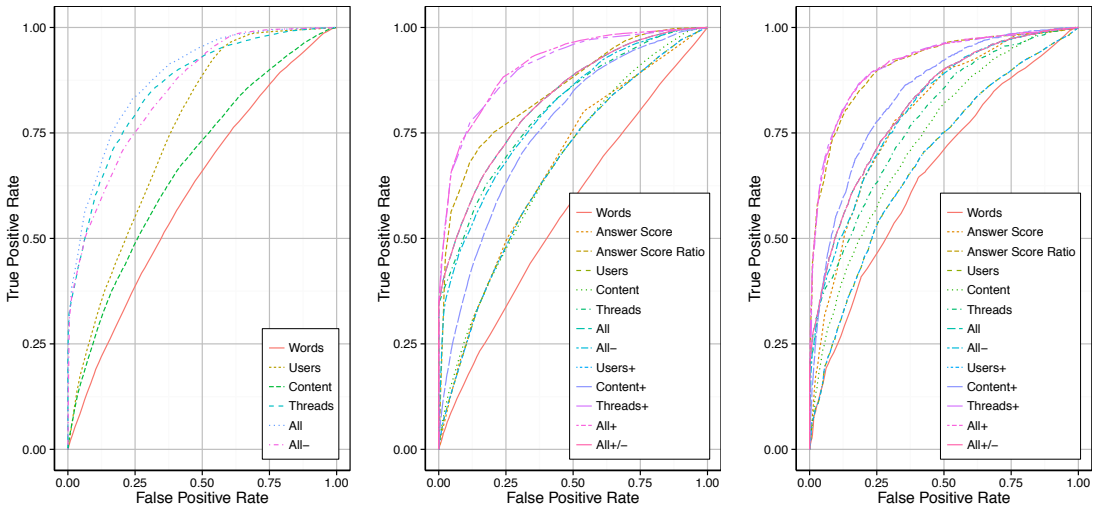
[193] Jeon et al. (2006); Agichtein et al. (2008)

such features are normally specially designed as a rating of content quality and usefulness.

Surprisingly, the results from all three datasets do not confirm previous research on the importance of content length for quality prediction. For each of the datasets, *precision* and *recall* is very low with an poor F_1 performance across each dataset (SCN: 0.419/SF: 0.552/CO: 0.606). This may be due to the difference of the data to those from literature which were taken from general Q&A communities such as Yahoo Answers [194] and the Naver community [195]. In particular, the SE and SCN communities studied in this thesis are mostly technical therefore, answers are likely to be more concise than in other communities and best answers may be not highly correlated with long answers.

[194] *Agichtein et al. (2008)*

[195] *Jeon et al. (2006)*



(a) SCN Forums (b) Server Fault (c) Cooking

Figure 8: Receiver Operating Characteristic (ROC) Curves for the *SCN Forums*, *Server Fault* and *Cooking* datasets using the *Multi-Class Alternating Decision Tree* classifier.

The SF and CO models trained on the *answer scores* highlight positive correlations between *best answers* and *scores*. Both datasets have high precision and recall with CO showing high F_1 results

with 0.704 and SF with 0.613 when using *answer scores*. Training the SE models on *Answer Score Ratios* shows even higher results with a F_1 of 0.792 for SF and 0.831 for CO. Overall, *answer score ratio* appear to be a good predicator for answer quality which shows that SF and CO collaborative voting models are effective. In particular, it shows that integrating the thread structure of Q&A communities by taking into account the relative voting proportions between answers (i.e. *scores ratio*) is a better approach than absolute *scores*.

Core Features Models: Here the focus is on the comparison of feature types (i.e. *users*, *content* and *threads*) and the impact of using the different features sets on the identification process. A model for each dataset and features set is trained. Results in Table 11 show that using the thread features introduced in this chapter increases accuracy in all three datasets over user and content features. Results also show that F_1 when combining all core user, content, and thread features was 3.3%, 2.4%, and 1.8% higher for SCN, SF, and CO respectively, than the best F_1 achieved when using these features sets individually.

Overall, when using all the core features (common to all datasets), SCN predictions perform better than SF (+4%) and CO (+7.9%). Predictions for SF and CO are both accurate with a respective F_1 of 0.751 and 0.724. This result is probably due to the similarity of both communities as they are based on the same platform. However, results in Table 11 show that F_1 with all core features is lower than the *Answer Score Ratio* by 5.2% for SF and 14.8% for CO. This reflects the value of this particular feature for *best answer* identification on such platforms.

Figure 9 shows the distributions of *best answer* (best) and *non-best answers* (normal) for posts length for all our datasets and answer scores for SF and CO. Best answers seem to likely be marginally shorter in SCN, and longer in SF and CO. This variation can be driven by the difference in community sizes and topics as well as external factors such as community policies (e.g. collaborative editing in SE).

Extended Features Models: Now the models are recomputed using extended *users*, *content* and *threads* feature sets. Remember that the extended features (Table 9) are only supported by SF and CO. No change in accuracy can be witnessed when extending the user features since they are the same as the core user features. However, F_1 increases by an average of 19.6% for SF and 16% for CO when extending content and thread features.

Table 11 shows that the F_1 for SF and CO when using all extended features combined (*All+* in 11) has increased by 14.6% and 16.1% for SF and CO respectively over using core features (*All* row in Table 11). This is mainly due to the addition of the *scores/ratings* based features. This observation is confirmed when comparing F_1 against the Answer Score Ratio model for SF and CO as each model does not improve much compared to the original baseline (+1% for SF, +2% for CO).

In general, thread features are consistently more beneficial than others for identifying *best answer*. When available, scoring (or rating) features improve prediction results significantly, which demonstrates the value of community feedback and reputation for identifying valuable answers.

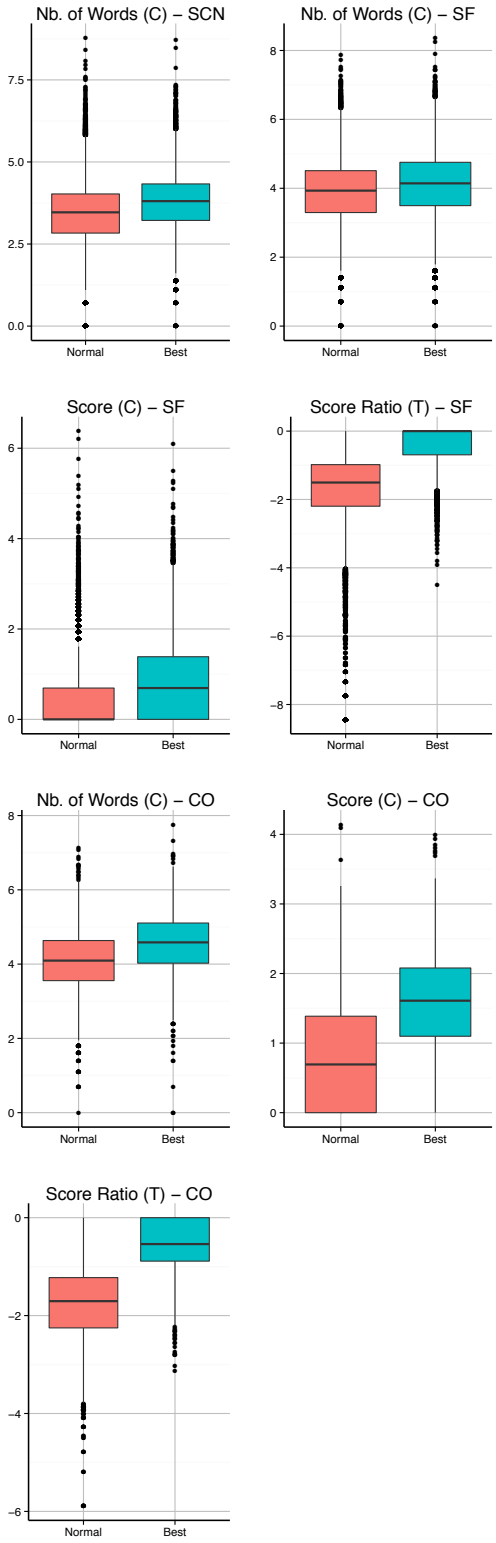


Figure 9: Box Plots representing the logarithmic distribution of different features and *best answer* for the *SCN Forums* (SCN), the *Server Fault* (SF) and *Cooking* (C) datasets.

Stable Features Models: Some features such as answer ratings may not be always available as the community needs time for rating answers as they are produced. Accordingly, the impact of the absence of such features globally is also studied.

For each dataset, it can be observed that the removal of non stable features impact negatively the identification of *best answer* (noted "-" in Table 11). However, for each dataset, the results are significantly higher than the baselines that are not based on answer ratings. For the SCN forums, the removal of evolving features lead to a decrease of F_1 by 4% compared with the usage of all the features whereas, SF and CO show a reduction of 10.9% and 15.2% in F_1 compared with the complete extended features sets.

As a summary, although the usage of non-stable features increase the identification of *best answer*, the lack of such information still provide decent predictions when compared with the baselines that are not based on answer ratings.

4.3.3 Results: Feature Selection and Best Models

The previous experiments showed that each feature has a contrasting impact on *best answer* identification. For instance, *answer length* seems to have little impact on identification results whereas *thread* features and answer ratings appear to be highly correlated to best answers. In order to better understand the impact of each individual features, it is necessary to analyse the importance of each predictor individually.

For each dataset, all the predictors are ranked using IGR with respect to the best answers labels. The top 15 are shown in Table 12.

	SCN		Server Fault		Cooking	
R.	IGR	Feature	IGR	Feature	IGR	Feature
1	0.0832	*Topic Rep. Ratio (T)*	0.1016	***Score Ratio (T)***	0.1552	***Score Ratio (T)***
2	0.0588	*Nb. Answers (T)*	0.0914	*Nb. Answers (T)*	0.0833	*Topic Rep. Ratio (T)*
3	0.0478	*Topic Rep. (U)*	0.0553	*Topic Rep. Ratio (T)*	0.0702	***Score (C)***
4	0.0368	*A. Succ. Ratio (U)*	0.0518	*Position (T)*	0.0619	*Nb. Answers (T)*
5	0.0337	*Reputation (U)*	0.0305	***Score (C)***	0.0535	*Position (T)*
6	0.0327	*Activity Entropy (U)*	0.0296	*Rel. Position (T)*	0.0446	*Answer Age (C)*
7	0.0317	*Nb. Bests (U)*	0.0223	*Answer Age (C)*	0.0354	*Nb. Bests (U)*
8	0.0316	*Question Ratio (U)*	0.0193	***Nb. Comments (C)***	0.0332	*Reputation (U)*
9	0.0312	*Answer Ratio (U)*	0.0161	*Q. Views (C)*	0.0315	***Nb. Comments (C)***
10	0.0278	*Rel. Position (T)*	0.0140	*A. Succ. Ratio (U)*	0.0313	*Post Rate (U)*
11	0.0277	*Z-Score (U)*	0.0090	*Z-Score (U)*	0.0307	*A. Succ. Ratio (U)*
12	0.0229	*Position (T)*	0.0088	*Nb. Posts (U)*	0.0269	*Nb. Posts (U)*
13	0.0152	*Nb. Answers (U)*	0.0081	*Community Age (U)*	0.0257	*Topic Entropy (U)*
14	0.0150	*Asking Rate (U)*	0.0078	*Reputation (U)*	0.0250	*Z-Score (U)*
15	0.0123	*Nb. Posts (U)*	0.0073	*Answering Rate (U)*	0.0243	*Term Entropy (C)*

Table 12: Top features ranked by Information Gain Ratio for the *SCN*, *Server Fault* and *Cooking* datasets. Type of feature is indicated by *U*/*C*/*T* for *User*/*Content*/*Thread*.

Core Features: The initial focus of the analysis is on the *core features set*. Table 12 shows that SCN's most important feature for *best answer* identification appears to be the *topical reputation ratio*, which also came high up the list with 3rd rank in SF and 2nd in CO. The *number of answers* also comes high in each dataset: 2nd for SCN and SF, and 4th for CO. Note that our training datasets only contained threads with *best answer*. Hence the shorter the thread is (i.e. fewer answers) the easier it is to identify the *best answer*. Figure 10 shows the correlations with *best answer* (best) and *non-best answers* (normal) for the top five features in each datasets.

For core features, SF, CO, and SCN have a generally similar mode of operation. However, SCN is less affected by *answer position* due to the difference of platform editing policies. SE favours small

threads whereas SCN does not. Such a difference leads to a better correlation of *number of answers* with *best answer* in SE.

According to Table 12, user features dominate the ranking, with some thread features amongst the most influential. Number of thread answers and historical activities of users are particularly useful (e.g. number and ratio of user's *best answer*). User reputation in SCN plays a more important role than in SF and CO, which is probably a reflection of the community policies that puts emphasis on members reputation.

In SCN, user activity focus seems to play a notable role (*topical reputation*, *answer and question ratios*, *activity entropy*, etc.). These features are further down the list for SF and CO.

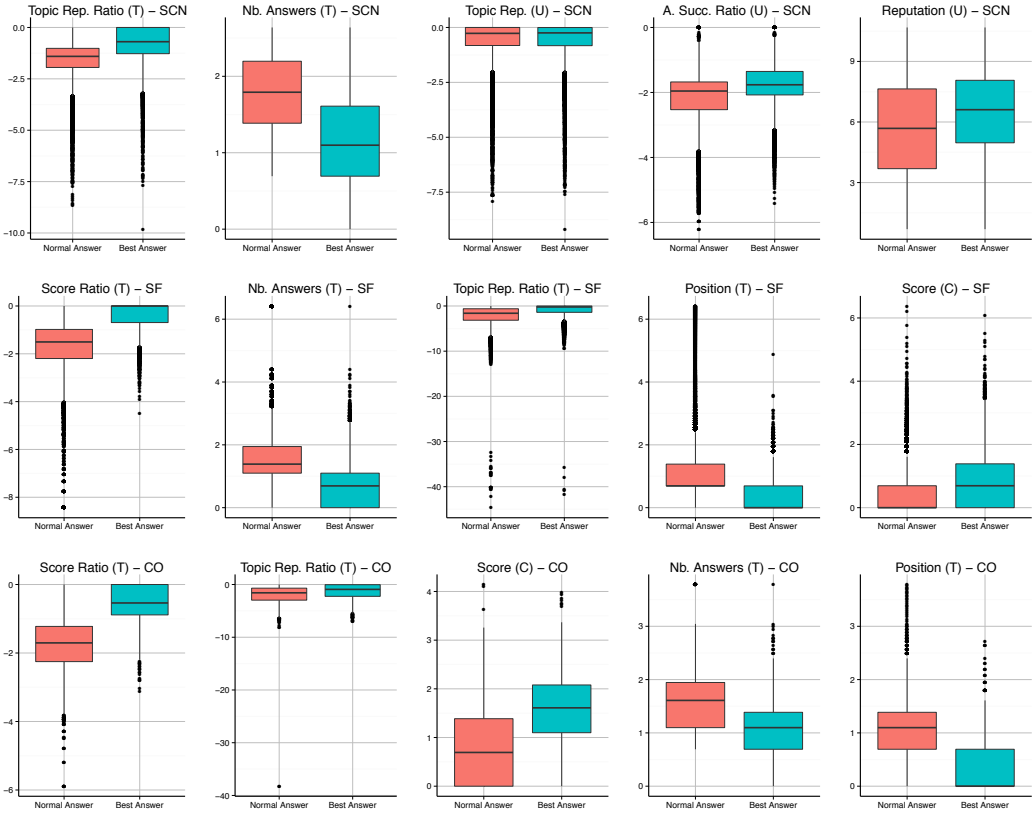


Figure 10: Box Plots representing the logarithmic distribution of the top five features for the *SCN Forums* (first row), the *Server Fault* (second row) and *Cooking* (third row) datasets.

Extended Features: The evaluation of extended features establishes the importance of *scores*. For SF and CO datasets, it is clear once again that the *score* features are the most important for identifying *best answer*.

SF has a *score ratio* IGR of 0.1016 and CO have IG score of 0.1552 representing respectively around +10% and +53.7% more gain than the second ranked feature.

As in the general model evaluation, thread features compare the *score* of a single answer with the score of other thread answers. The higher the ratio, the better the answer. Note that the selection of *best answer* in SF and CO is left to the user who posted the question, who may or may not consider the scores given by the community or general site visitors.

Stable and Evolving Features: The ranking obtained by calculating the IGR of stable features does not show important differences compared with just two non stable features listed in the top 15 features (*number of answers* and *relative answer position*). For the other datasets, higher impact can be observed as *score* based features are removed. In any case, the top feature listed for each dataset is the *topical reputation ratio*. This result confirm that the *reputation ratio* based on the previous reputation of individuals in particular topics is a consistent indicator of *best answer* for each of the studied datasets.

4.4 DISCUSSION

The main goal of this chapter is to introduce the different *best answer* identification models that are improved upon in the following chapters as well as the sets of features that are used in the rest of this thesis.

Although different types of Q&A communities tend to have different characteristics, goals, and behaviours, the *best answer* identification models studied in this thesis show state of the art results with an $F_1 > 0.75$ for the studied datasets. The difference between the communities studied in this thesis and those used in previous investigations (Chapter 3) makes it difficult to compare directly the findings without a broad base of experimentation.

[196] Jeon et al. (2006); Agichtein et al. (2008)

For instance, compared to previous works,[196] *content length* appears to not be correlated with *best answer*. Nevertheless, the analysis done in this chapter was performed on three different communities that vary in size, topic as well as underlying platform. The three communities were selected for giving more scope and depth to those findings. Since the studied communities bear much similarity in terms of type, goals, and properties, it can be argued that the findings of this chapter could be transferred across communities that are similar to the one studied.

From the results, it appears that identifying *best answer* becomes more important the longer the threads are. Therefore, it might be worth focusing such analysis on threads with more than one answer.

It is worth mentioning than in SCN, SF, and CO datasets, the median number of answers per thread was 5,3, and 4 respectively, with averages of 13, 8.5, 5.

For the SF and CO communities, the ratings given by community members to existing answers appear to be good predictors of *best answer*. Although only the authors of questions can currently pick the *best answer* in the studied communities, their choices seem to be positively correlated with those of the public. The results showed that the accuracy of using public ratings for *best answer* selection can be marginally improved further when other features are considered. SCN currently lacks community ratings. Interestingly, SCN has migrated itself to the Jive Engage platform[197] in 2012. Jive offers many social features, including collaborative rating of answers as discussed in chapter 2.

[197] *Jive Software*, http://jivesoftware.com.

The particular effectiveness of *thread* features such as *score ratio* show that features that take into account the structure of Q&A communities are good predictor of *best answer*. Such result partially confirm the hypothesis about the development of optimisation methods that use the thread structure of Q&A communities (H1.1) and prompt the generalisation of such approach to all the features studied in this chapter. Different generalisation of the *thread* features are proposed and evaluated in the following chapter as the structural design methodology (RQ1.1) is evaluated.

Although, models of *question complexity*, *user maturity* and *contribution effort* are introduced and studied in the later chapters (Chapter 6 and Chapter 7), it worth mentioning that features like user *reputation* and to some extent *answer age* are well ranked in Table 12.

Since user *reputation* and *answer age* can be related to *user maturity* and *contribution effort*, it can be expected that the qualitative features introduced in chapter 6 and chapter 7 will improve the *best answer* identification model presented in this chapter and confirming the hypotheses about the applicability of qualitative design to *best answer* identification (H1.2).

4.5 SUMMARY

This chapter presented different models for automatically identifying *best answer* by using a wide selection of *user*, *content* and *thread* features. Some of those features were common across all three communities, and some were community-specific. This chapter provided a state of the art *best answer* identification model with 78.1% F_1 with SCN community, 83.3% with SF and 83.8% with CO.

[198] Jeon et al. (2006); Agichtein et al. (2008)

Contrary to previous work,[198] it was found that *answer length* seems to be uncorrelated with *best answer*. This difference may be due to different factors: 1) For instance, the editing policy of SE favour concise answers instead of long answers; 2) The studied communities are mostly technical websites and focused on particular topics, and; 3) Each community encourage non-conversational and opinion answers therefore length may be not a good predictor of *best answers*.

It was also discovered that *best answer* in communities that support community-based answer ratings (i.e. SF and CO) can be identified

much more accurately, with over 0.8 F_1 using this feature alone *(answer score ratio)*.

Unfortunately, answer ratings may not always be available as users need time to rate community posts. In this context, an important decrease of F_1 was observed particularly for SF and CO ($> 8\%$) when evolving features are omitted. This result shows that being able to measure or predict the rating of answers may be beneficial for *best answer* identification when such information is unavailable.

The thread-based features proved to be very influential for *best answer* identification in all three communities confirming the importance of structural design in the creation of *best answer* predictors. To some extent features related to *question complexity*, *user maturity* and *contribution effort* showed some promise confirming the importance of modelling such features more accurately.

In the following chapters, the presented *best answer* models are reused for evaluating the different hypotheses presented in chapter 1. In particular, the following chapter explores different structural optimisation techniques based on the thread-like structure of Q&A communities and generalise the *thread* features presented in this chapter.

5

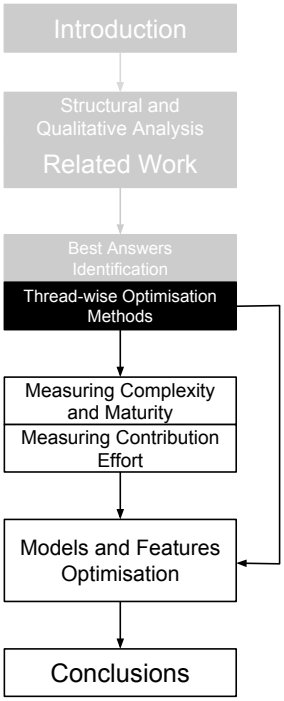
THREAD-WISE OPTIMISATION METHODS

In the previous chapter it was shown that it is possible to identify relatively accurately *best answers* by using binary classifiers coupled with *user*, *content* and *thread* features.

This chapter investigates two distinct optimisation techniques for improving the accuracy of the models discussed in the previous chapter (Chapter 4). First, thread based normalisation methods are introduced for improving the accuracy of predictions by introducing a systematic normalisation approach that normalise predictors by taking into account relations between features relations.

Second, LTR models are applied for ranking answers within a question thread in order to identify *best answers*. Compared to the models presented in chapter 4, better results are obtained for each of the three datasets studied in this thesis. These results show that structural design helps the identification of *best answers* (RQ1.1).

This chapter is divided in seven sections. First, the importance of feature optimisation and the need of thread based optimisation and LTR models is discussed before different normalisation approaches



are presented. In the third section normalisation methods are compared with non-normalised models before the accuracy of LTR models is evaluated. Finally, the results are discussed and the chapter summarised.

5.1 INTRODUCTION

Although previous research such as the one introduced in chapter 4 shows that *best answers* can be identified relatively accurately using binary classifiers, a few different methods can be used for improving the results of the previous models. In this thesis two different methods are proposed: 1) A structural design approach that use the structure of Q&A communities to optimise prediction models and features (RQ1.1), and; 2) A qualitative design approach that propose the identification of important features based on user beliefs (RQ1.2). This chapter investigates the first research question (RQ1.1) and if "the thread-like structure of Q&A communities can help the automatic identification of *best answers*" (H1.1).

In the previous chapter, it was shown that *thread* features are useful as they present relations between answers of a same thread. Similarly, other works such as Gkotsis et al.[199] on the usage of normalised shallow features demonstrated that taking into account feature value order between answers of a same thread dramatically improved *best answer* identification.

[199] *Gkotsis et al. (2014)*

Building on those previous contributions and the observations made in chapter 2, this chapter proposes to generalise the thread features

of the previous chapter and the ranking method proposed by Gkotsis et al.[200]. Besides generalising these techniques, additional normalisation methods that can be applied automatically to any datasets that have thread like structures such as query search results in IR are also proposed.

[200] *Gkotsis et al. (2014)*

Due to the similarity between search results in IR and question threads the usage of LTR models for predicting *best answers* in a given thread is also investigated since such models can be used for ranking the most likely *best answers* in a thread and may be more accurate than the binary classifiers usually used for *best answers* identification.

Accordingly, in this chapter, methods for enhancing *best answer* predictions are proposed by the use of a) thread-based feature normalisation approaches, and; b) LTR methods. Consequently, the main contributions of this chapter are:

1. Introduce a systematic approach for normalising features based on answering threads.
2. Compare the applicability of four different thread based normalisation methods: min/max normalisation, normalised min/max normalisation, order normalisation and normalised order normalisation.
3. Evaluate the performance of a pointwise LTR approach for identifying *best answers*.
4. Investigate the impact of rank based features on *best answers* binary classifiers and pointwise LTR models.
5. Investigate if structural design improves *best answer* identification (RQ1.1).

5.2 THREAD-WISE OPTIMISATIONS FOR PREDICTING BEST ANSWERS

As observed in the previous chapter (Chapter 4), thread features are highly associated with *best answers*. Therefore, such observation can be generalised in different ways for improving *best answers* identification models.

The following section presents two different techniques for integrating the thread structure of Q&A communities into *best answer* models.

5.2.1 *Thread-wise Normalisation*

Feature normalisation has been used in different ML settings in order to deal with features that have outliers and ensure that ML algorithms consider independent features equally during the learning and prediction phases. A typical approach used for normalising features is based on the min/max formula[201] that scale numerical variables between 0 and 1. Unfortunately, such approach requires the knowledge of the boundaries of the studied variable which may shift when additional data is analysed. For example, in Q&A communities, the *reputation* of users has no boundaries therefore min/max normalisation is not easily applicable. Another issue is the use of global minima and maxima instead of their local counterparts (i.e. community extrema instead of answering threads extrema).

[201] *The min/max normalisation function $MM(x,X)$ that returns a normalised value of a given feature value $x \in X$, where X is the observed set of all the values of a particular feature is given by:*

$$MM(x,X) = \frac{x - \min X}{\max X - \min X}$$

Another approach is to use sigmoid normalisation[202] as it does not

[202] *The sigmoid normalisation function $Sig(x)$ that returns a normalised value of a given feature value x is given by:*

$$Sig(x) = \frac{1}{1 + e^{-x}}$$

requires definite knowledge about the upper and lower bounds of the studied variable. However even though sigmoid methods are more suitable, they may reach their maximum value too quickly when the range of observable values are high. In this context, the logistic normalisation function[203] can be preferred but it still needs to be parametrised properly to fit a particular variable for obtaining the best results (i.e. sigmoid midpoint and steepness value).

[203] *The logistic function normalisation function $LSig(x)$ that returns a normalised value of a given feature value x between 0 and 1, where k represents the curve steepness and x_0 is the midpoint x-value is given by:*

$$LSig(x) = \frac{1}{1 + e^{-k(x-x_0)}}$$

As observed in the previous chapter, the usage of taking into account the relative values of a given feature within a thread helps the identification of *best answers* (i.e. the local relations for a given features are more useful than community wide values).

Calculating features ratios such as *score ratios* improve the ability to identify *best answers* compared to the scores of individual answers. Following this observation different normalisation methods can be extrapolated. As a consequence all features become thread features as they represent the comparison of predictors values across threads.

In the following section, different normalisation are proposed. In particular, the ordering approach used by Gkotsis et al.[204] is generalised and extend by normalising the orders across question threads. Each method is separately evaluated in section 5.4.

[204] *Gkotsis et al. (2014)*

Min/Max Normalisation The min/max feature normalisation approach is based on the min/max normalisation function applied to a particular feature of a given thread. Consequently, each feature value is normalised using the maximum and minimum of that feature for that particular thread. Formally, the min/max normalisation function $TN_{mm}(v_i, V_{f,t})$ normalise a value v_i of a given feature

$f \in F$ within a question thread $t \in T$ where $v_i \in V_{f,t}$ and $V_{f,t}$ contains all the values of f for the thread t. As a result, the min/max normalisation function $TN_{mm}(v_i, V_{f,t})$ is defined as:

$$TN_{mm}(v_i, V_{f,t}) = \frac{v_i - \min(V_{f,t})}{\max(V_{f,t}) - \min(V_{f,t})} \tag{16}$$

For example for an answering thread with a feature that takes the values $V = \{31, 10, 5\}$, the corresponding normalised values are $V_{mm} = 1, 0.19, 0\}$.

Normalised Min/Max Normalisation The normalised min/max normalisation method is based on the min/max normalisation approach and extends it by dividing the results by the length of the thread. Accordingly, the normalised min/max normalisation function normalised min/max normalisation $TN_{mmrat}(v_i, V_{f,t})$ is given by:

$$TN_{mmrat}(v_i, V_{f,t}) = \frac{TN_{mm}(v_i, V_{f,t})}{\|V_{f,t}\|} \tag{17}$$

For instance for an answering thread with a feature that takes the values $V = \{31, 10, 5\}$, the corresponding normalised values are $V_{nnrat} = \{0.33, 0.06, 0\}$.

Order Normalisation The order normalisation approach generalises the approach presented by Gkotsis et al.[205] to any feature. Each feature is given a rank between 1 and the length of a question thread. If the value is the smallest for a given feature in a thread, it is given a value of one. If it is the highest value, it is given a value that equals the length of the thread. Intermediate values are valued similarly.

[205] Gkotsis et al. (2014)

Using the same notation as the proportional normalisation method, the order normalisation function $TN_{order}(v_i, V_{f,t})$ is designed to return the index of a given value v_i in a given list of values ordered by decreasing order $V_{f,t}$. The returned value is bounded according to $\|V_{f,t}\|$ (i.e. $[1, \|V_{f,t}\|]$).

For example for an answering thread with a feature that takes the values $V = \{31, 10, 5\}$, the corresponding normalised values are $V_{order} = \{1, 2, 3\}$.

Normalised Order Normalisation The normalised order method is based on the previous order normalisation approach. However, instead of returning absolute numbers, it divides the results by the length of the thread so that across threads, the normalisation is always bounded between zero and one. Given the order normalisation function $TN_{order}(v_i, V_{f,t})$, the normalised order function is given by:

$$TN_{orat}(v_i, V_{f,t}) = \frac{TN_{order}(v_i, V_{f,t})}{\|V_{f,t}\|} \tag{18}$$

For instance for an answering thread with a feature that takes the values $V = \{31, 10, 5\}$, the corresponding normalised values are $V_{orat} = \{\frac{1}{3}, \frac{2}{3}, \frac{3}{3}\}$.

Adaptive Features Normalisation Some features do not necessarily vary within threads such as the *number of answers* or the *number of question views* therefore, normalising them will not be useful as such predictors only vary across threads. In order to account for such type of features automatically, the variance of values within threads for the whole dataset for a given feature is calculated. If

the variance is zero and remains constant between all the threads, the features is not normalised. Otherwise, the feature is normalised with one of the previous functions.

Depending on the classification algorithm used for identifying *best answers*. It may be useful to drop such features as they are invariant within a thread. However, in this thesis, the used classification algorithm is based on decision trees. As a consequence, the variation across thread may help the algorithm to distinguish sub-classification settings (e.g. when there is only one answer or when there is multiple answers).

5.2.2 *Learning To Rank Models*

Best answers identification depends only on finding a *best answer* within a question thread. As a consequence, *best answer* identification can be seen as a LTR task where the goal is to associate the highest ranked answer as the *best answer*.

LTR approaches follow three distinct methods for learning a ranking function that helps the ranking of relevant documents given a list of documents. The different methods are: 1) Pointwise ranking; 2) Pairwise ranking, and; 3) Listwise ranking.[206]

[206] *Liu (2009)*

Pointwise Ranking: The pointwise approach is based on the classification of single documents. Each documents is directly evaluated on the given ranking function and an absolute relevance score is returned that gives the relevance and absolute position of the document.

Pairwise Ranking: The pairwise approach does not assume absolute relevance labels but instead focus on the comparison of document pairs. Documents are ranked according to their preference order score obtained from a ranking function that compare document pairs.

Listwise Ranking: The listwise approach directly treat document lists as learning instances and learn a ranking function that directly return ranked lists rather than individual rank for each list documents. Therefore, instead of reducing ranking a classification task, learning is achieved directly on document lists: an entire ranked list is treated as a learning instance.

Although many different approaches exist for ranking documents,[207] the approach used in this chapter is based on a pointwise method as they tend to perform better than other approaches on similar datasets and where only one document need to be identified from a subset of documents[208] (i.e. a unique *best answer*). Moreover, by using a pointwise method, it is possible to reuse the models presented in the previous chapter making it easier to compare the prediction results between this chapter and the previous one.

[207] *Liu (2009)*

[208] *Burel et al. (2015a,b)*

Since the goal is to identify the answer that is most likely to be the *best answer* for a question, the likelihood value that an answer is a *best answer* is used. Similarly to the previous chapter, the *Alternating Decision Tree* algorithm is chosen as it provided good results for predicting *best answers*.

Given the computed likelihood $P(a|\mathscr{L}(f)$ of a classifier $\mathscr{L}$ (in this case the *Alternating Decision Tree* algorithm) for a given answer

$a \in A_i$ and a vector of corresponding features values $f \in F_i$ the *best answer* can be identified by finding the answer with the highest likelihood in A_i:

$$Best(A_i, F_i) = \underset{a \in A_i, f \in F_i}{\operatorname{argmax}} P(a|\mathscr{L}(f)) \tag{19}$$

The proposed pointwise ranking method is much simpler than conventional LTR models where multiple items need to be labelled with different values. Nevertheless, although out of scope of this thesis the same algorithm could be used for ranking answers in a thread according to their likelihood to be the *best answer*.

5.2.3 Features List

In order to compare the advantages and disadvantages of using thread normalisation and LTR models for identifying *best answers*, all the features introduced in chapter 4 are reused. The feature types presented in the previous chapter are also reused and baseline features are distinguished (*number of words*, *answer score* and *answer score ratio*) from the *user*, *content* and *thread* features. Finally, the *extended* and *stable* features sets are also studied. The list of used features is reproduced in Table 13.

	Features Set		
Type	Core Features Set (28)	Extended Features Set (31)	Current Features (24)
User	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)
Content	*Answer Age, Number of Question Views, Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (6)	***Score**, Answer Age, Number of Question Views, **Number of Comments**, Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy*. (8)	*Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (4)
Thread	*Number of Answers, Answer Position, Relative Answer Position, Topical Reputation Ratio.* (4)	***Score Ratio**, Number of Answers, Answer Position, Relative Answer Position, Topical Reputation Ratio*. (5)	*Answer Position, Topical Reputation Ratio.* (2)

Table 13: List of features and features categories.

5.3 THREAD-WISE NORMALISATION METHOD SELECTION

Before evaluating how thread normalisation impacts the identification of *best answers*, it is important to determine what normalisation approach is the most likely to provide the best results. In order to find the approach that works best for the datasets studied in this thesis, the average IG for *best answer* identification of all the features presented in the previous section is compared for each dataset and with the three normalisation methods (Table 14).

The average IG of the normalised feature generally shows an increase compared with the non normalised features except for the min/max and normalised min/max methods. In particular, the order normalisation approach provides the highest gains with an average

IG of 0.1210. The normalised order normalisation also provides good results with an IG of 0.0851 whereas the min/max and normalised min/max approaches do not improve IG (0.0421 and 0.0503).

Those results show that in general normalisation approaches can improve *best answer* identification compared with the absence of normalisation. The order normalisation method seems to provide the best result, therefore it is retained as the normalisation approach applied in the rest of this chapter.

5.4 Best Answers Identification using Thread-wise Normalisation

In the previous chapter different predictors of *best answers* were introduced. Although good results were obtained, the models were not optimised by taking into account the structure of Q&A communities even though it can be expected that optimisations methods such as feature normalisation can increase the accuracy of ML tasks.

As part of the structural design methodology proposed in this thesis (RQ1.1), thread-wise optimisation methods are selected due to the particular structure of Q&A communities (H1.1, chapter 2). The following experiments aim at evaluating the impact of thread normalisation on *best answer* identification by highlighting how each feature impacts prediction accuracy for different datasets and platforms. The goal is also to determine if "structural optimisation techniques improve automatic *best answer* identifications and if so how" (H1.1).

		Normalisation Method			
Dataset	Original	Min/Max	Norm. Min/Max	Order	Norm. Order
SCN	0.0642	0.0447	0.0446	0.1105	0.0891
Server Fault	0.0476	0.0331	0.0480	0.1387	0.1044
Cooking	0.0654	0.0485	0.0581	0.1137	0.0620
Average	0.0591	0.0421	0.0503	0.1210	0.0851

Table 14: Average IG for each dataset and different thread normalisation approach for identifying *best answers*.

5.4.1 *Experimental Setting*

In this experiment, the impact of order normalisation on *best answer* identification is compared for each of the datasets used in this thesis. An approach similar to the one discussed in section 4.3.1 of the previous chapter is applied. However, the 10-folds stratified cross-validation is performed differently as full answering threads are required for identifying *best answers* in order to apply the LTR model discussed in section 5.2.2.

Rather than dividing each dataset based on all the datasets answers, each dataset is split by answering thread by keeping the thread lengths proportional to the dataset so that the proportion of *best answers* and non *best answer* is similar to the standard 10-folds stratified cross validation applied in the previous chapter. This method is designed so that the new results can be compared with the results discussed in the previous chapter.

As in the previous chapter, the precision (P), recall (R) and the harmonic mean F-measure (F_1) are reported as well as the area under the Receiver Operator Curve (ROC) measure. The experiment is done using the *Alternating Decision Tree* algorithm and the normalised and non normalised results are compared. The features that

are the most relevant are also discussed by reporting the IGR of individual features.

5.4.2 *Results: Model Comparison*

In order to compare the impact of thread normalisation with the non-normalised features, the results for both the normalised and non-normalised results are reported as a different cross-validation method is used compared to the previous chapter. The results are listed in Table 15.

A look at the results shows that the *Precision*, *Recall*, F_1 and *AUC* are similar to the results reported in the previous chapter. Both results are similar since both folding approaches are stratified and keep the same proportion of non-*best answers* and *best answers*. A paired $t-$test comparing the non-normalised features sets for each datasets with the results of the previous chapter and the thread folding approach show no significance in F_1 with a $p-$value of 0.1513, therefore the results between each folding approaches are comparable.

Baseline Models: The normalisation approach used in this chapter shows a relatively good performance of the *number of words* feature. For the non-normalised features, the F_1 for SCN is 0.419, 0.552 for SF and 0.606 for SF. For the order normalised version, the F_1 is 0.713 (+41.4%) for SCN, 0.715 (+24.1%) for SF and 0.715 (+15.3%) for SF. This results shows that the length of answers can identify *best answers* when the relative length of answers is used.

		SCN Forums				Server Fault				Cooking			
Model	Features	*P*	*R*	F_1	*AUC*	*P*	*R*	F_1	*AUC*	*P*	*R*	F_1	*AUC*
Std.	Words	0.500	0.360	0.419	0.611	0.519	0.590	0.552	0.566	0.566	0.651	0.606	0.652
	Answer Score	-	-	-	-	0.592	0.635	0.613	0.672	0.692	0.719	0.705	0.795
	Answer Sc. Ratio	-	-	-	-	0.783	0.801	0.792	0.847	0.824	0.839	0.831	0.908
	Users	0.565	0.674	0.615	0.755	0.593	0.632	0.612	0.669	0.592	0.661	0.624	0.687
	Content	0.550	0.656	0.599	0.673	0.592	0.637	0.614	0.674	0.625	0.687	0.654	0.737
	Threads	0.727	0.788	0.756	0.860	0.720	0.745	0.733	0.807	0.653	0.773	0.708	0.783
	All	0.753	0.811	0.781	0.883	0.725	0.777	0.750	0.829	0.687	0.764	0.724	0.817
	All-	0.712	0.799	0.753	0.851	0.693	0.765	0.727	0.802	0.686	0.739	0.711	0.809
	Users+	-	-	-	-	0.593	0.632	0.612	0.669	0.592	0.661	0.624	0.687
	Content+	-	-	-	-	0.681	0.692	0.686	0.761	0.734	0.755	0.744	0.843
	Threads+	-	-	-	-	0.820	0.842	0.831	0.908	0.828	0.854	0.841	0.912
	All+	-	-	-	-	0.823	0.844	0.833	0.912	0.821	0.849	0.835	0.913
	All±	-	-	-	-	0.725	0.777	0.750	0.829	0.687	0.764	0.724	0.817
Norm.	Words	0.713	0.713	0.713	0.763	0.727	0.727	0.727	0.771	0.715	0.715	0.715	0.765
	Answer Score	-	-	-	-	0.826	0.826	0.826	0.863	0.853	0.853	0.853	0.884
	Answer Sc. Ratio	-	-	-	-	0.826	0.826	0.826	0.863	0.853	0.853	0.853	0.884
	Users	0.725	0.799	0.760	0.855	0.717	0.765	0.740	0.811	0.682	0.761	0.719	0.791
	Content	0.701	0.792	0.744	0.796	0.727	0.763	0.744	0.815	0.681	0.738	0.708	0.790
	Threads	0.665	0.847	0.745	0.819	0.721	0.739	0.730	0.804	0.650	0.761	0.701	0.771
	All	0.772	0.807	0.789	0.877	0.731	0.778	0.754	0.833	0.723	0.766	0.744	0.824
	All-	0.771	0.805	0.788	0.876	0.726	0.778	0.751	0.830	0.719	0.766	0.742	0.822
	Users+	-	-	-	-	0.717	0.765	0.740	0.811	0.682	0.761	0.719	0.791
	Content+	-	-	-	-	0.824	0.829	0.826	0.901	0.847	0.855	0.851	0.913
	Threads+	-	-	-	-	0.826	0.826	0.826	0.903	0.853	0.853	0.853	0.903
	All+	-	-	-	-	0.831	0.828	0.829	0.910	0.848	0.855	0.851	0.914
	All±	-	-	-	-	0.731	0.778	0.754	0.833	0.723	0.766	0.744	0.824

Table 15: Average answer *Precision*, *Recall*, F_1 and *AUC* for the *SCN Forums*, *Server Fault* and *Cooking* datasets for different feature sets and extended features sets (marked with +) and reduced features sets (marked with -) using the *Alternating Decision Tree* classifier and thread order normalisation.

Similarly to previous observations, the *answer score* and *answer score ratios* features are very good predictors of *best answers*. In particular, by using thread normalisation, both features become very good predictors with an average F_1 of 0.826 for SF and 0.853 for CO.

Looking at the distribution of baseline normalised features (Figure 11), it can be observed that answers that are longer than the other thread answers are more likely to be *best answers*. Similarly higher score means better answers. Such results are similar to the results discussed in the previous chapter.

Overall the normalisation approach benefits a lot the *answer score* and *answer score ratios* features. This observation confirms that relational features (i.e. thread features) and *score* based metrics are very good *best answer* predictors.

Core Features Models: Lets now focus on the core feature types (i.e. *users*, *content* and *threads*) for analysing the impact of feature sets on the identification process. Similarly to the *baseline* features, higher precision/recall compared to the analysis performed in the previous chapter is found.

For the SCN and SF communities and the non-normalised features, the least useful features are content feature (F_1: SCN:0.599, SF:0.614) followed by the user features (F_1: SCN:0.615, SF:0.612) and thread features (F_1: SCN:0.756, SF:0.733). These results are similar to what was obtained when the global cross-validation method was applied (Chapter 4).

Although a general increase in F_1 appears compared to the non-normalised features, the impact of feature set is largely different

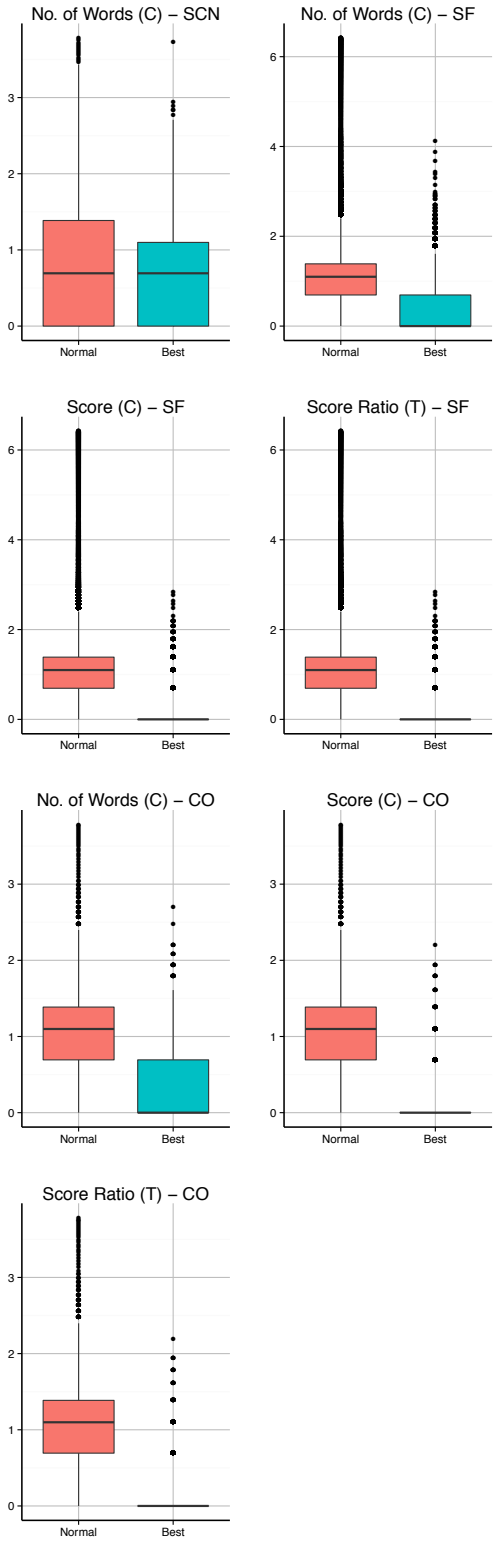


Figure 11: Box Plots representing the logarithmic distribution of different order normalised features and *best answers* for the *SCN Forums* (SCN), the *Server Fault* (SF) and *Cooking* (C) datasets.

as all features become relational. In this situation, the *thread* features are the least efficient (with no observable real difference compared with the non-normalised features) followed by the *user* features ($+20.3\% F_1$ on average compared with the non-normalised features) and the *content* features ($+21\% F_1$ on average compared with the non-normalised features).

Since all features become *thread* features because of the normalisation method, it is somehow expected that they perform lower than the other feature sets as the *thread* set has significantly less features than the other sets. Even thought, the difference between F_1 medians of the *users* and *content* feature set is minimal ($< 1\%$), it appears that content features play a higher role when relations between answers are taken into account. This result confirms the findings of Gkotsis et al.[209] that shallow content features are efficient for distinguishing quality and low quality answer within threads. These findings also highlight that reputation information about user may be only useful when used globally (i.e. distinguishing quality answers at the community level) rather than locally (i.e. distinguishing quality answers at the thread level).

[209] Gkotsis et al. (2014)

Using all non-normalised features give better result than only relying on individual feature sets. Such results are similar to what was observed in the previous chapter. When the order normalisation is used, results highlight similar patterns with *score ratios* giving high accuracy. In general, it appears that the *all* normalised feature perform better than the *all* non-normalised feature (F_1: SCN:0.789, SF:0.754, CO:0.744 Vs. F_1: SCN:0.781, SF:0.750, CO:0.724).

Extended Features Models: The main difference between using *core* and *extended* features is the presence of *scores*. The presence of such scores makes evident the importance of scores as *content* and *thread* feature become the best feature sets compared to the *user* set when using normalised features (Table 15). When not using normalised features the results are similar to the previous chapter where *thread* features produced better performances than the other sets.

Looking at the combined feature sets (Table 15), it appears that the results are not significantly different when using or not using thread normalisation, when normal features are used and when order normalisation is applied. However, the thread normalisation approach gives better precision with a median F_1 increase of 2%.

Stable Features Models: Since some features may not always be available such as answer ratings, the analysis is also performed on stable feature sets(Table 15). The removal of evolving features shows an important drop in F_1 across each dataset. Although the drop in accuracy seems important, results are still relatively accurate with an $F_1 > 0.7$ for each dataset and when using normalised and non-normalised features. Consistently with what was previously observed, the usage of normalisation gives an increase in precision and recall.

As a summary, it appears that thread normalisation approaches improves *best answer* identification. A tailed pared t-test between the results of the non-normalised models and the normalised models for each datasets and features sets confirms such a relation with a

$p-$value of $2.817e-05$. On average, an increase in F_1 of $+5.3\%$ is observed compared to the non normalised method presented in the previous chapter.

5.4.3 Results: Feature Selection

Following the last experiment, the second analysis evaluates the importance of individual features based on their order normalisation. In order to infer what normalised features are the most important, the IGR of the top features is calculated for each of our datasets in Table 16. Then, the results can be contrasted with the rankings of non-normalised features discussed in chapter 4 (Table 12).

Core Features: First, the focus is on the *core feature* set. Table 16 shows that SCN's most important feature appears to be the *ratio of answers* posted by answers authors. Such feature seems not as important for the other datasets (ranked $12th$ for SF and $> 15th$ for CO). The *user reputation* feature seems important for each dataset (ranked $3rd$ for SCN, $9th$ for SF and $8th$ for CO) meaning that the amount of knowledge users have may influence *best answer* identification positively. As illustrated by Figure 12, for SCN, *best answers* are correlated with the most knowledgeable users (i.e. higher reputation). The *term entropy* feature is generally well ranked (ranked $10th$ for SCN and $5th$ for SF and CO). Looking at the distribution of *term entropy* for SF and CO (Figure 12), it appears that the answer that have more diverse vocabulary are more likely to be *best answers*. This observation shows that *best answers*

may be more detailed compared to the other answers of the same thread.

Compared to the ranking of the non-normalised features observed in the previous chapter, *user* features also play a dominant role. However, *topic reputation* does not seem to be an important feature in this context, meaning that this feature only helps when distinguishing *best answers* in a global context. The user tendency to answer questions becomes useful when used for distinguishing *best answers* within threads as users that are focused on answering seem to provide better answers (Figure 12).

Extended Features: When observing extended features, the score measures are the most important (+40% IGR and +54% IGR for SF and CO compared to the second ranked features). Both *score ratios* and *scores* are ranked at the same position as both metrics become the same when normalised. Such results are largely comparable to the results of chapter 4 where *score ratios* where ranked the highest.

Compared with the previous chapter rankings, the *number of comments*, which only exists in the CO and SF datasets, appear important as a low amount comments correlate with good answers (Figure 12). For example users may use comment sections to point necessary changes to users for creating a better answer. Therefore, the relative amount of comments may be a good indicator of *best answers*.

Stable and Evolving Features: The ranking of *stable* features shows important differences compared to the *core* features rankings as

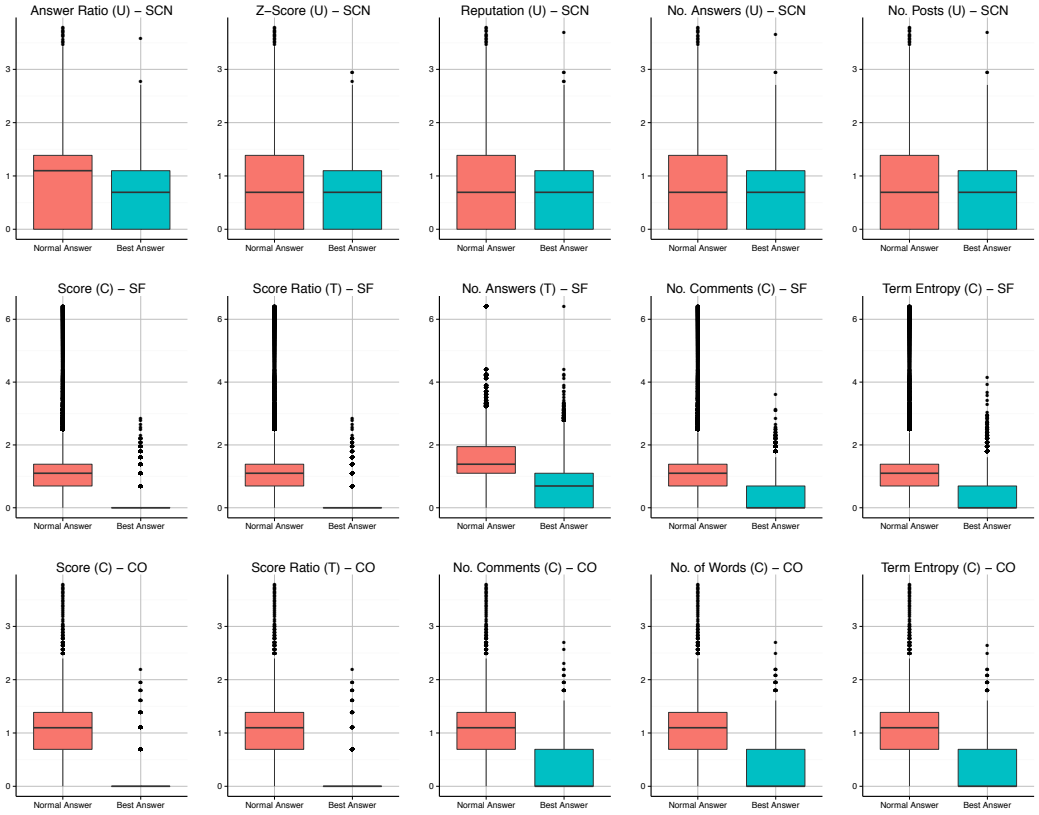


Figure 12: Box Plots representing the logarithmic distribution of different the top five order normalised features for the *SCN Forums* (first row), the *Server Fault* (second row) and *Cooking* (third row) datasets.

only the *number of answers* and the *relative answer position* are missing for SCN. However the other datasets miss more critical features such as the answer *scores*. The top ranked predictors appear to be the same as the *core* features for SCN. For SF and CO, the best feature is the *term entropy*. As previously seen it confirms that *best answers* have a relatively more diverse vocabulary compared to the other answers of a thread.

Compared to the non-normalised features it can be seen again that topic reputation is not as important when using relational features as answers are compared between each other in a given thread.

As a summary, a difference between non-normalised and normalised rankings can be observed. Although, ratings remain highly correlated with quality content in each case, it seems that content features are more important when used as relations rather than when used globally. This shows that the impact of features is highly different when used locally (i.e. when comparing within a thread) compared to globally (i.e. when comparing across all the answer of a community).

	SCN		Server Fault		Cooking	
R.	IGR	Feature	IGR	Feature	IGR	Feature
1	0.0763	*Answer Ratio (U)*	0.1532	***Score (C)***	0.1732	***Score (C)***
2	0.0747	*Z-Score (U)*	0.1532	***Score Ratio (T)***	0.1732	***Score Ratio (T)***
3	0.0683	*Reputation (U)*	0.0914	*Nb. Answers (T)*	0.0793	***Nb. Comments (C)***
4	0.0681	*Nb. Answers (U)*	0.0809	***Nb. Comments (C)***	0.0686	*Nb. of Words (C)*
5	0.0644	*Nb. Posts (U)*	0.0765	*Term Entropy (C)*	0.0674	*Term Entropy (C)*
6	0.0607	*Nb. Bests (U)*	0.0754	*Nb. of Words (C)*	0.0651	*Nb. Bests (U)*
7	0.0588	*Nb. Answers (T)*	0.0628	*A. Succ. Ratio (U)*	0.0650	*A. Succ. Ratio (U)*
8	0.0546	*A. Succ. Ratio (U)*	0.0538	*Q. Succ. Ratio (U)*	0.0623	*Reputation (U)*
9	0.0537	*Answering Rate (U)*	0.0527	*Reputation (U)*	0.0619	*Nb. Answers (T)*
10	0.0536	*Term Entropy (C)*	0.0522	*Nb. Bests (U)*	0.0505	*Answering Rate (U)*
11	0.0527	*Nb. of Words (C)*	0.0500	*Nb. Posts (U)*	0.0502	*Z-Score (U)*
12	0.0510	*Community Age (U)*	0.0495	*Answer Ratio (U)*	0.0498	*Nb. Posts (U)*
13	0.0474	*Topic Rep. (U)*	0.0482	*Nb. Answers (U)*	0.0497	*Nb. Solved (U)*
14	0.0474	*Topic Rep. Ratio (T)*	0.0477	*Nb. Solved (U)*	0.0485	*Nb. Answers (U)*
15	0.0466	*Post Rate (U)*	0.0476	*Question Ratio (U)*	0.0483	*Nb. Questions (U)*

Table 16: Top order normalised features ranked by Information Gain Ratio for the *SCN*, *Server Fault* and *Cooking* datasets. Type of feature is indicated by U/C/T for User/-Content/Thread.

5.5 BEST ANSWERS IDENTIFICATION USING LEARNING TO RANK MODELS

5.5.1 Experimental Setting

Similarly to the previous experiment, the usage of LTR models for identifying *best answers* as well as the impact of order normalisation on LTR results is evaluated in order to determine if algorithms that take into account the structure of Q&A communities help the identification of *best answers* (H1.1). In this experiment, a thread-based stratified 10-folds cross-validation approach is used and the *Precision*, *Recall*, *AUC* and the F_1 measure is reported.

5.5.2 Results: Model Comparison

For comparing the accuracy of the LTR approach, predictions accuracy of the LTR approach for each dataset and for the normalised and non-normalised features are reported in Table 17. The results can be compared with the results shown in Table 16.

Baseline Models: The LTR models show better performance over the non-normalised and non-LTR approaches for the *baseline* features (Table 17). LTR alone is almost as good as the order normalisation (method for the *number of words* feature median $F_1 = 0.708$ vs. median $F_1 = 0.718$). However, when using both LTR and order normalisation together, the *number of words* median F_1 increase by 1% ($F_1 = 0.715$) making it as efficient as using order normalisation features alone.

		SCN Forums				Server Fault				Cooking			
Model	Features	*P*	*R*	F_1	*AUC*	*P*	*R*	F_1	*AUC*	*P*	*R*	F_1	*AUC*
LTR	Words	0.708	0.684	0.696	0.708	0.721	0.711	0.716	0.721	0.719	0.697	0.708	0.720
	Answer Score	-	-	-	-	0.832	0.815	0.823	0.832	0.860	0.835	0.847	0.860
	Answer Sc. Ratio	-	-	-	-	0.829	0.813	0.821	0.829	0.857	0.835	0.846	0.857
	Users	0.744	0.701	0.722	0.744	0.717	0.709	0.713	0.717	0.699	0.687	0.693	0.700
	Content	0.704	0.671	0.688	0.704	0.716	0.704	0.710	0.716	0.739	0.715	0.727	0.740
	Threads	0.757	0.736	0.746	0.757	0.678	0.671	0.674	0.678	0.678	0.678	0.678	0.678
	All	0.795	0.780	0.787	0.795	0.738	0.727	0.732	0.738	0.737	0.731	0.734	0.737
	All-	0.756	0.733	0.744	0.756	0.720	0.714	0.717	0.720	0.726	0.726	0.726	0.726
	Users+	-	-	-	-	0.717	0.709	0.713	0.717	0.699	0.687	0.693	0.700
	Content+	-	-	-	-	0.823	0.812	0.817	0.823	0.848	0.832	0.840	0.848
	Threads+	-	-	-	-	0.828	0.815	0.822	0.828	0.856	0.830	0.843	0.856
	All+	-	-	-	-	0.843	0.829	0.836	0.843	0.852	0.841	0.846	0.853
	All±	-	-	-	-	0.738	0.727	0.732	0.738	0.737	0.731	0.734	0.737
LTR	Words	0.713	0.713	0.713	0.713	0.727	0.727	0.727	0.727	0.715	0.715	0.715	0.715
(Norm.)	Answer Score	-	-	-	-	0.826	0.826	0.826	0.826	0.853	0.853	0.853	0.853
	Answer Sc. Ratio	-	-	-	-	0.826	0.826	0.826	0.826	0.853	0.853	0.853	0.853
	Users	0.750	0.749	0.749	0.750	0.700	0.700	0.700	0.700	0.697	0.697	0.697	0.697
	Content	0.713	0.713	0.713	0.713	0.726	0.726	0.726	0.726	0.713	0.713	0.713	0.713
	Threads	0.730	0.723	0.726	0.730	0.622	0.627	0.624	0.622	0.631	0.613	0.622	0.631
	All	0.780	0.780	0.780	0.780	0.730	0.730	0.730	0.730	0.727	0.727	0.727	0.727
	All-	0.780	0.780	0.780	0.780	0.729	0.729	0.729	0.729	0.722	0.722	0.722	0.722
	Users+	-	-	-	-	0.700	0.700	0.700	0.700	0.697	0.697	0.697	0.697
	Content+	-	-	-	-	0.827	0.827	0.827	0.827	0.851	0.851	0.851	0.851
	Threads+	-	-	-	-	0.826	0.826	0.826	0.826	0.853	0.853	0.853	0.853
	All+	-	-	-	-	0.832	0.832	0.832	0.832	0.853	0.853	0.853	0.853
	All±	-	-	-	-	0.730	0.730	0.730	0.730	0.727	0.727	0.727	0.727

Table 17: Average answer *Precision*, *Recall*, F_1 and *AUC* for the *SCN Forums*, *Server Fault* and *Cooking* datasets for different feature sets and extended features sets (marked with +) and reduced features sets (marked with -) using the LTR and thread order normalisation.

For the other baselines, similar results are obtained compared to non-normalised and normalised features with *score* based predictors performing the best.

In general, LTR is better than using non-normalised features but it performs best when used with normalisation techniques. However for baseline features the same results can be obtained when using only normalised features meaning that LTR may be not as good as feature normalisation.

Core Features Models: Using the different feature sets, it appears that LTR provides better results compared to non-normalised features with a similar F_1 (Figure 17). However, there is a lower accuracy compared to the usage of order normalisation (with an average F_1 for all the features of SCN:0.718/SF:0.716/CO:0.693 instead of SCN:0.749/SF:0.738/CO:0.709). When using both LTR with normalisation, the accuracy becomes similar to order normalisation alone.

Each individual feature set generates better results compared to the non normalised features but results are lower than the normalisation approaches described in the previous sections. As previously observed, the combination of the LTR approach with the normalisation method is close to the application of order normalisation alone. Therefore, in general, simple normalisation seems to be enough for improving *best answers* identification in Q&A communities.

Extended and Stable Features Models: Best predictions are observed when using simple order normalisation without using the LTR approach (Figure 17). However LTR remains a better performer than the usage of non-normalised features alone.

As previous observation has shown, answer ratings improve results while focusing on stable features reduces predictions efficiency.

As a summary, using only normalised features is enough for obtaining good predictions. However, LTR models proved rather efficient compared to non-normalised features particularly when applied to small communities. The advantage of simple normalisation compared to LTR may be explained by a disparity between the high

likelihood of *non-best answers* within a thread and the likelihood of actual *best answers*. It would be interesting in future work to investigate if *best answers* remain highly ranked even when they do not have the highest likelihood to be *best answers* when using LTR.

Although the LTR approach does not perform better than order normalisation for identifying *best answers*, compared to the non normalised approach described in the previous chapter a tailed paired $t-$test between each feature sets and approach show that the LTR methodology does improve identification results with a $p-$value of 0.00013. However the advantage of order normalisation over both LTR and LTR with feature normalisation is not significant with a the respective $p-$values of 0.41 and 0.40.

5.6 DISCUSSION

In order to improve the results presented in the previous chapter, different methods based on the hypothesis that the "thread-like structure of Q&A communities can help the automatic identification of *best answers*" (H1.1)).

Although this work is similar to previous research,[210] this contribution varies significantly as the concept of thread normalisation was formalised and different normalisation techniques were introduced. Besides the previous contribution it was also shown that although not as efficient as order normalisation, the LTR approach is relatively good at identifying *best answers*.

[210] *Gkotsis et al. (2014)*

Despite not being able to compare directly these results to Gkotsis et al.[211] work due to the small difference in the evaluation method,

[211] *Gkotsis et al. (2014)*

it is likely that the additional features help compared to the reduced feature set that was used in their research. Indeed, the results shown in chapter 4 are more accurate compared to the approach proposed Gkotsis et al.[212] thanks to the *score* features.

[212] *Gkotsis et al. (2014)*

The results show the important difference highlighted when using non-normalised features and relational features obtained (i.e. thread normalisation). Although features used in this chapter and the previous chapter are the same, the thread normalisation showed that content features are good locally (i.e. at the tread level) even though they are not useful when used globally. This result shows the importance of normalisation as features with limited utility become relevant thanks to simple transformations

Surprisingly, LTR did not provide results as good as expected even though precision and recall improved compared to a non-normalised setting. Such result may be due to the simple likelihood-based maximisation approach used for identifying *best answers*. It would be interesting in future research to evaluate how good are the predictions by using traditional LTR metrics such as the MEAN RECIPROCAL RANK (MRR). Other future work should also explore more complex LTR methods such as pairwise and listwise ranking models.

In general, the usage of thread-wise optimisation techniques proved to improve results compared to the model introduced in the previous chapter, therefore, the structural optimisation methodology appear to improve *best answer* prediction in the dataset studied. As a result, it can be argued that structural optimisation helps *best answer* identification (RQ1.1).

5.7 SUMMARY

Two different approaches for improving such types of models were proposed based on the hypothesis that "the thread-like structure of Q&A communities can help the automatic identification of *best answers*". Such hypothesis was formulated by following the structural design methodology investigated in this thesis (RQ1.1, Chapter 2). First, different feature normalisation methods based on the threaded nature of Q&A communities were proposed. Second, an LTR model was applied for improving results quality.

Order normalisation was the most useful normalisation technique even though there was no significant difference compared to the LTR approach. The analysis also showed that to some extent, LTR approaches may be used for identifying *best answers*. Although on average across all the features sets only a improvement of $+5.3\%$ was reported compared to the usage of non-normalised features, this improvement is consistent across all the datasets and significant ($2.817e-05$) even thought the improvement over the previous best result is not important.

The normalisation method highlighted the importance of content features when used at the thread level such as *term entropy*. This observation shows that some features become only useful when used as relations. As the best model for predicting *best answers* is obtained when only using thread normalisation, the following experiments on *best answer* identification reuse this model.

In this chapter it was shown that structural optimisation improves *best answer* identification (RQ1.1). In the following chapters new

features for improving the accuracy of the models presented in this chapter are introduced in order to evaluate the hypothesis linked to the qualitative design approach presented in this thesis (RQ1.2). Following the community study in chapter 2, the research in the next chapters focus on two different features: 1) The maturity of users, and 2) the contribution effort of answerers.

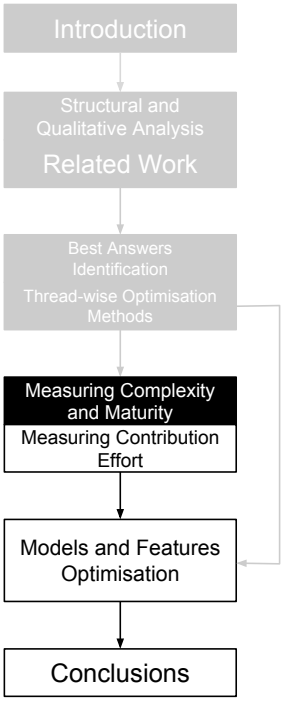
Part III

QUALITATIVE DESIGN AND BEST ANSWER IDENTIFICATION

6

MEASURING COMPLEXITY AND MATURITY

The measure of question complexity and user maturity presented in this chapter is designed to identify user expertise (RQ1.3). The question complexity information can be used in different ways to model user ability and expertise that complement the score based measures presented in chapter 4. In particular, it is considered that "*knowledgeable users are users that are able to answer and ask complex questions*" (H1.3). This hypothesis forms the base of the proposed maturity metric.



The SF community is used for creating the question complexity measures that model the level of expertise required to answer a question. The experiments conducted in this five parts chapter show that question complexity depends on both the length of involvement and the level of contributions of the users who post questions within their community. Although the findings highlight the difficulty of

automatically identifying question complexity, it appears that complexity is influenced by the topical focus and the length of community involvement of askers. Following the identification of question complexity, a measure of maturity is presented and the evolution of different topical communities is presented. The results show that different topical communities show different maturity patterns. Some communities show a high maturity at the beginning while others exhibit slow maturity rate. The study of user maturity also show that users with high reputation are more likely to contribute to complex questions (H1.3). Therefore it can be claimed that maturity can be used for representing user expertise (RQ1.3).

Finally, in order to analyse the two other datasets of this thesis, a metric approximating question complexity called omega (Ω) is proposed based on the complexity model created on SF.

6.1 INTRODUCTION

Besides studying the impact of structural design on *best answer* identification (RQ1.1), this thesis also investigates if user beliefs about what makes quality answers can be exploited in order to design a better predictor of *best answers* (RQ1.2). Based on different user studies (Chapter 2), two features are investigated: 1) Question complexity and user maturity; a measure of user knowledge (RQ1.3), and; 2) Contribution effort; a measure of user reactivity (RQ1.4). In order to evaluate if such features improve the identification of *best answer*, it is first required to design these features. This chapter investigates the measurement of question complexity and user maturity.

The well being of Q&A communities depends on many factors (Chapter 2). One of such factor is the interests of user in *learning new things* (Chapter 2, Q14: "*Why do you participate in this online community?*"). Based on such observation it appears that the measurement of learning ability in Q&A communities can be useful for evaluating the status of such communities.

In this chapter, the concept of maturity is introduced as a proxy measure of user knowledge and defined as: *the ability of contributors to respond to or ask complex questions* (H1.3). Maturity can be used by community managers to monitor the progression of their community as well as identify sub-communities and topics that are highly specialised. Managers can also employ maturity to identify communities that are becoming less mature and react accordingly.

Understanding the maturity of individuals and the complexity of questions may be used as an additional proxy measure of expertise and help the identification of *best answer* by linking *best answers* to answering ability. Although it is expected that complex questions should increase over time, empirical work in validating this assumption is lacking. In particular, despite the importance of measuring community maturity, no measure has been proposed for measuring both question complexity and community maturity of Q&A communities.

To address the above issue, question complexity is defined based on the hypothesis that *the complexity of a question is influenced by the previous activities of its asker*. Different factors involved in question complexity are analysed and a model that can be used for analysing the maturity of SF, a Q&A community is developed. Based on the model created for SF, a complexity metric that can

be applied to the other communities studied in this thesis is created. The analysis of this chapter focuses on the SF dataset as complexity annotations are required in order to distinguish the complex questions from the easy questions. The annotations are performed on SF as annotators familiar to the topics discussed in the SF community were available when doing the annotation process. The contributions of this chapter are summarised below:

1. Introduce a definition of question complexity and validate the hypothesis that question complexity increases with askers' community involvements.
2. Study the influence of features relating to askers, answerers, questions and answers on question complexity prediction.
3. Introduce a complexity metric that can be used on arbitrary communities that does not have complexity annotations and evaluate its predictive power for the SF community.
4. Introduce the concept of community maturity, a measure of community knowledge and specialisation.
5. Investigate the evolution of community maturity in SF and demonstrate that community maturity is influenced by topical dynamics.
6. Investigate if users with high reputation are more likely to have high maturity (H1.3).

6.2 Defining Question Complexity and Community Maturity

The concept of maturity is defined on top of question complexity where maturity is measured as the proportion of complex questions asked at a given time. As such, the first step is to define question complexity since community maturity is measured based on question complexity.

6.2.1 *Question Complexity*

Question complexity can be viewed as *the level of knowledge required for understanding a particular text* and define question complexity as:

Definition 6.1 (Question Complexity)**.** *Question complexity is a value representing the difficulty and level of expertise required for answering a question.*

Measuring complexity is a difficult task since it depends on the notion of expertise and knowledge. Although the level of knowledge embedded in a particular piece of text can be somehow estimated using vocabulary analysis and community ratings, this work proposes to measure question complexity by using a number of different factors.

When the community moderators of the SCN forums were surveyed (Chapter 2),[213] it was found that they believe that power

[213] Rowe et al. (2011a)

users are more interested in complex questions rather than easy ones; meaning that such users are more likely to ask and answer more complex questions over time. In this chapter the relation between time and question complexity is summarised in the following hypothesis:

Hypothesis 1 (Temporality)**.** *For a given user, question complexity increases as a function of time and participation. The longer a user is actively involved in a community, the more complex her questions are.*

Apart from *temporality*, additional factors that could also potentially affect question complexity are identified. In particular, facets like the number of questions asked by a user (*Enquiry*), their activity levels (*Commitment*), the number of questions that have been solved (*Accomplishment*) and their focus on particular domains of interest (*Focus*) can influence question complexity. Additional hypotheses concerning question complexity are formulated below:

Hypothesis 2 (Enquiry)**.** *For a given user, question complexity increases with the number of questions asked. The more a user asks or answers, the more likely her questions will become more complex or her answers will target complex questions.*

Hypothesis 3 (Commitment)**.** *For a given user, question complexity increases with her activity levels. The more frequently a user is involved in a community, the more complex are her questions.*

Hypothesis 4 (Accomplishment)**.** *For a given user, question complexity increases with the number of her questions that have been answered before. The more a user finds answers to some questions, the more likely she is likely to asks more complex questions in the future.*

Hypothesis 5 (Focus)**.** *For a given user, question complexity depends on her topical focus. If a user focused a lot on a narrow set of topics and interests, it is more likely she will become knowledgeable in those specific topical areas and hence ask more complex topic-specific questions.*

6.2.2 *Community Maturity*

Question complexity can be seen as a good measure of expertise since it is more probable that knowledgeable users ask more complex questions than others. Following this assumption and our hypotheses that question complexity is dependent on *time*, *enquiry*, *commitment*, *accomplishment* and *focus*, community maturity is defined as a measure indicating the level of specialisation and knowledge of a community as follows:

Definition 6.2 (Community Maturity)**.** *Community Maturity is a value representing the level of knowledge and specialisation achieved by a community. A more mature community focuses on more complex questions whereas a community less mature has simpler and*

less focused questions. Such a maturation process should particularly affect communities that have long term contributors.

6.3 Features Relating to Question Complexity

Measuring and validating question complexity and community maturity requires the identification of relevant features. In this section, features relating to four different groups are studied: *askers* and *answerers* (*users*), and; *questions* and *answers* (*content*).

Some of the features used for identifying complex questions are adapted from the metrics defined in the previous chapters. Accordingly, only the new feature that were not introduced previously are discussed in detail below.

6.3.1 Asker Features

Asker features capture the characteristic of users who post questions. Below is a list of such features used in this chapter.

- *Community Age (Experience)*: It measures the length of user's involvement in a community, i.e., how many days an asker has been active in the community. This feature represents the concept of *temporality* expressed in Hypothesis 1. This feature is the same as the *community age* user feature introduced in chapter 4.

- *Community Age Difference*: The difference between an asker's community age and the mean of all her answerers' community ages. This feature represents the experience gap between an asker and her answerers.
- *Number of Questions (Enquiry)*: The number of questions posted by an asker. This represents the concept of *enquiry* expressed in the Hypothesis 2.
- *Number of Answers*.
- *Asking Rate (Asker Commitment)*: Average number of questions an asker posts per day. This measure represents the concept of *commitment* expressed in Hypothesis 3. This feature is the same as the *asking rate* user feature introduced in chapter 4.
- *Answering Rate*.
- *Ratio of Successfully-Answered Questions*.
- *Ratio of Question Successfully Answered by Others (Accomplishment)*: This feature is the same as the *question success ratio* user feature introduced in chapter 4. This feature represents the concept of *accomplishment* expressed in Hypothesis 4.
- *Normalised Question Topic Entropy (Focus)*: Calculates the concentration of a user's questions across different topics. This feature represents the concept of *focus* expressed in Hypothesis 5. This feature is similar to the *topic entropy* user feature introduced in chapter 4 but only considers questions instead of all the users' posts.
- *Normalised Answer Topic Entropy*: Calculates the concentration of a user's answers across topics. It is similar to the *normalised question topic entropy*.

- *Average Number of Replies per Question*: Average number of replies received by a user's questions.
- *Average Number of Question Views*: Average number of views received by a user's questions.
- *Z-score*.[214]
- *Reputation*.

[214] Zhang et al. (2007)

6.3.2 Answerer Features

Answerer features are similar to asker features but rather than being calculated on individuals, they are derived in an aggregated manner at the thread level. Taking the feature "*Community Age*" as an example, for a given question, the *Community Age* of answerers is represented by the mean (and standard deviation) of the answerer *Community Age* value in a given thread. The same 14 features mentioned in section 6.3.1 are used for answerers and the mean and standard derivation for each of these features is derived for any given question.

6.3.3 Question Features

Question features, which represent the attributes of questions, are listed below:

- *Number of Views*.
- *Number of Words*.
- *Readability with Gunning Fog Index*.
- *Readability with Flesch-Kincaid Grade*.

– *Existing Value*:[215] The measure $V_d(q_i)$ of a question q_i represents the value of this question derived solely from its answers $a_{q_{i_j}} \in A_{q_i}$. In a typical Q&A online community, users can vote on answers posted. $score(a_{q_{i_j}})$ is used to denote the vote received by answer $a_{q_{i_j}}$, and $status(a_{q_{i_j}})$ to denote whether the answer has been flagged as "*best answer*" or "helpful answer" ($status(a_{q_{i_j}}) = 2$), or not ($status(a_{q_{i_j}}) = 1$). The value of $V_d(q_i)$ is defined by the following equation:

[215] Pal and Konstan (2010)

$$V_d(q_i) = \min\left(5, \sum_{j=1}^{|A_{q_i}|} score(a_{q_{i_j}}) + status(a_{q_{i_j}})\right) \quad (20)$$

– *Status*: Represents the current state of a question as having a *best answer* or not.
– *Number of Answers*: Number of answers received by a particular question. This feature is the same as the *number of answers* thread feature introduced in chapter 4.
– *Favourites*: Number of times a question has been bookmarked by users.
– *Score*.
– *Informativeness*: The informativeness $I_d(q_i)$ of a question q_i essentially measures how many novel words occur in question q_i given all the previous questions Q. A question with more new words is more likely to be novel and hence potentially more complex. Let $|T_{q_i}|$ denote the total number of words appeared in q_i, $|t_{q_{i_j}}|$ denote the occurrence frequency

of word t_j in q_i, $|Q_{t_j}|$ denote the number of previous questions also containing word t_j, the informativeness measure $I_d(q_i)$ of question q_i is defined as:

$$I_d(q_i) = \sum_{j=1}^{|T_{q_i}|} \frac{|t_{q_{i_j}}|}{|T_{q_i}|} \times \log \frac{|Q|}{|Q_{t_j}|+1} \tag{21}$$

– *Cumulative Term Entropy*.

6.3.4 Answer Features

The final set of features represents properties of a particular thread (i.e. all answers relating to a question). Answer features aim to capture general characteristics of answers in a given thread. Similarly to answerer features, answer features are represented by the mean and standard deviation of feature values aggregated from individual answers.

The following question features as defined in section 6.3.3 are also used as answer features using their standard deviation and means: *Number of Words*, *Score*, *Informativeness* and *Cumulative Term Entropy*.

In addition, five additional features are added for answers:

– *Average Number of Elapsed Days*: Calculates for each answer the number of elapsed days between the date the question was posted and the date the answer was provided. This feature is related to the *answer age* content feature introduced in chapter 4.

- *Elapsed Days First*: Represents the elapsed days between the date the question posted and the date the first answer provided.
- *Elapsed Days Last*: Represents the elapsed days between the date the question posted and the date the last answer provided.
- *Number of Comments Mean*: Represents the average number of comments received by each answer of a thread. This feature is related to the *number of comments* content feature introduced in chapter 4.
- *Score*: The total number of points received by a question's answers.

6.4 MEASURING QUESTION COMPLEXITY

Before analysing the evolution of maturity in the SF community, the previous hypotheses need to be validated and a model for question complexity prediction needs to be created.

6.4.1 *Experimental Setting*

Four different tasks are performed. First, a subset of the SF dataset is manually annotated and a gold standard is generated. Second, the hypotheses concerning the relation between user contributions and question complexity are validated using tailed $t-$tests. Third, a logistic regression models from the annotated data is trained using the features defined in section 6.3 in order to automatically identify complex questions. Fourth, for improving the question complexity

prediction performance of the regression models, four different feature selection methods are used and additional models are trained on feature subsets.

Hypothesis validation is performed using tailed $t-$tests. The different complexity models are evaluated with 10-folds cross validation. For each model, the precision (P), recall (R) and the harmonic mean F-measure (F_1) as well as the Area Under the Receiver Operator Curve (AUC) measure are reported. Similarly, feature selection is verified with 10-folds cross validation.

6.4.2 *Question Complexity Annotation*

For verifying the hypotheses, a set of SF question pairs from a group of users are annotated by identifying which question is more complex. Each pair contains questions from the same user. Rather than selecting users randomly, users that have a sustained community involvement are only selected, i.e., asked many questions (relating to *Enquiry*) over a long time period (relating to *Commitment* and *Experience*). Users that receive valid answers (*Accomplishment*) and are focused on particular topics (*Focus*) are also selected. These selection criteria favour users that are more susceptible to learn from the answers they received and therefore increase their knowledge and potentially raise more complex future question about a particular topic.

A ranking score RS_u is designed for each user u by jointly considering the five main factors which could potentially influence question complexity presented in section 6.2. For the five factors, exp_u denotes the user *experience*, enq_u represents the number of questions

asked by a user (*enquiry*), com_u denotes the user *commitment* to the community, acc_u represents the user accomplishment in obtaining answers to his questions, and foc_u denotes the user topical *focus*. These variable values are normalised using min/max normalisation and a log transformation is performed before combining them into a ranking function defined below:

$$RS_u(exp_u, enq_u, com_u, acc_u, foc_u) = exp_u \times enq_u \times com_u \times acc_u \times (1 - foc_u) \quad (22)$$

The users are ranked by RS_u and the top hundred users are selected. For each user, their posted questions are ordered by posting time and question pairs are created by selecting one question from the top 10% and another one from the bottom 10%. The resulting set contains 510 question pairs.

The annotation was performed by three annotators who have background in system administration. Each annotator was presented with the question pairs in random order and was asked to select the most complex question from the two displayed (i.e. *Which question is the most complex? Left or Right*). The Kappa inter-annotator agreement for the annotation results showed a very low inter-annotator agreement ($\kappa = 0.146$). This is mostly due to the difficulty in evaluating some particular question pairs and some annotators' unfamiliarity with the topics of certain questions (e.g., one annotator might be more familiar with Windows servers while others have better understanding of Unix systems). In order to alleviate this issue and obtain a valid gold standard, the final selected dataset is only composed from the annotated question pairs that have more than 75%

Table 18: Statistical hypothesis testing using a t-test for each annotator and for the gold standard.

			P-value		
Annotator	Pairs	Latest Mean	$H_1 : \mu \neq \mu_0$	$H_1 : \mu > \mu_0$	$H_1 : \mu < \mu_0$
A	510	0.6039216	2.151e-06***	1	1.075e-06***
B	510	0.5764706	5.219e-04***	9.997e-01	2.61e-04***
C	510	0.5509804	2.115e-02**	9.894e-01	1.058e-02*
Gold	220	0.65	5.638e-06***	1	2.819e-06***

Signif. codes: p-value < 0.001 *** 0.01 ** 0.05 * 0.1 .

agreement. The gold standard contains 220 pairs out of a total of 510 pairs.

6.4.3 *Hypothesis Testing*

In this section, the validation of the main hypothesis (Hypothesis 1) stated in section 6.2 is performed. The statistical significance of the annotation results is computed by calculating the p-values for each annotator and also for the gold standard. For testing the main hypothesis, the null hypothesis H_0 and the corresponding testing hypothesis H_1 are defined . The null hypothesis considers that "the complexity of the questions asked by a user is independent on her length of involvement in a community". The testing hypothesis states that "the complexity of the questions asked by a user is dependent on her length of involvement in a community". Although the main hypothesis states that question complexity increases with time, this assumption is relaxed in the testing hypothesis so tailed t-tests can be performed($H_1 : \mu \neq \mu_0$, $H_1 : \mu < \mu_0$ and $H_1 : \mu > \mu_0$). The results are reported in Table 18.

TIME DEPENDENCY ($H_1 : \mu \neq \mu_0$): On average, long established users' questions are identified as complex (Table 18).

Such results show that questions contributed by established users seem to be more complex than their earlier questions. All sided tests are statistically significant. Particularly, the p-value associated with the gold standard shows a high significance level ($5.638 \times 10^{-06} \ll 0.001$). This result strongly rejects the null hypothesis. Therefore, *question complexity depends on the community age of users*.

COMPLEXITY DECREASES WITH TIME ($H_1 : \mu < \mu_0$): The results show that complexity does not decrease with time as the null hypothesis fails to be rejected with p-value close to 1. Therefore, *question complexity does not decrease with the community age of askers*.

COMPLEXITY INCREASES WITH TIME ($H_1 : \mu > \mu_0$): The results show a low p-value ($2.819 \times 10^{-06} \ll 0.001$). Given those results, it can be argued that *question complexity increases with the community age of askers* thus validating the main hypothesis.

6.4.4 Question Complexity Prediction

With the annotated question pairs, a classifier is trained on the various features discussed in section 6.3 in order to predict whether a given question is complex or not.

The classifier is trained on each individual question that form the gold standard. Because the pairs that share low inter-annotator agreement were discarded, the risk of having ambiguously annotated questions is averted. In addition, although each question is annotated in pairs, what makes a question complex is independent from the question pairings. As a consequence, high and low question complexity remains consistent even if question pairs are divided. Such setting make it possible to train automatic question complexity classifiers on the annotated data.

As previously mentioned, a 10-folds cross validation is performed and precision (P), recall (R), the F-measure (F_1) and the Area Under the Receiver Operator Curve (AUC) measure are reported (Table 19). Only the results from the logistic regression model are presented since it gives the best results compared to Naive Bayes, Support Vector Machine, or Maximum Entropy.

Baseline Models: Since the main hypothesis states that question complexity increases with an asker's community age, a baseline logistic regression model is trained using *asker's age* as the only feature. Also, intuitively, a question is likely to be more complex if it contains more words. Hence, another baseline model based on the *number of words* contained in the question is created. It can be observed from Table 19 that *asker's community age* appears to be a better predictor than *number of words* since it gives a better performance than the latter. Therefore, it appears that the length of a question is highly associated with question complexity.

Features Type Models and Complete Model: A logistic regression model is also trained separately for each type of features presented in Section 6.3: 1) asker features; 2) answerer features; 3) question features; and 4) Answer features. The results in Table 19 show that using either *asker* or *answerer* features gives similar results and only marginally outperforms the baseline model trained on *asker's community age* only. Using *question* features gives slightly worse results compared to user features. *Answer* features do not seem to help in question complexity prediction as they are only slightly better than random guessing. Combining all the features ("All" in Table 19) gives only a small performance gain compared to the models trained on individual type of features (+2.8% F_1 on average).

	Complexity Model			
Features	*P*	*R*	F_1	*AUC*
Asker's Community Age	0.596	0.594	0.593	0.591
Question Words	0.484	0.484	0.477	0.478
Askers	0.602	0.601	0.601	0.660
Answerers	0.601	0.601	0.601	0.610
Questions	0.593	0.592	0.591	0.576
Answers	0.527	0.525	0.513	0.539
All	0.606	0.606	0.605	0.621
Info. Gain Ratio	0.641	0.641	0.640	0.660
CFS	0.648	0.647	0.647	0.664
Feature Drop	0.630	0.629	0.628	0.664
CFS+PC+FD	0.623	0.622	0.621	0.649
Ω (Complexity Measure)	0.571	0.821	0.654	0.638

Table 19: Average *Precision*, *Recall*, *F*1, *AUC* for the SERVER FAULT dataset for different feature sets using Logistic Regression and the Omega Metric.

Features Selection and Best Model: In this section, feature selection is performed in order to find out which set of features is most

important to question complexity prediction. Three different methods for feature selection are selected: IGR, CFS and finally features are ranked by dropping an individual feature one by one from the full regression model and accounting for the drop in F_1 (ablation test).

FEATURES SELECTION: The rankings obtained from each feature selection method are listed in Table 20. Consistent with what have been observed from the previous model results, *user* features are the most significant in determining question complexity representing on average 73.3% of the top ten features. The very top features are consistent across all the feature selection methods. In particular, *asker* features like the *ratio of successfully-answered questions* (A. Success Ratio (UQ)) and the *normalised question topic entropy* (Q. Topics. Ent. (UQ)) appear among the top four features which implies that they play very important roles in predicting question complexity.

The distributions of the top 5 features selected by IGR and CFS are also shown in the box plots in Figure 14. It can be observed that askers who posted a high proportion of correct answers in the past are more likely to ask complex questions (see the box plot for A. Success Ratio (UQ)). On the other hand, askers who have specific topical interests tend to post hard questions (see the box plot for Q. Topics. Ent. (UQ)). A question feature, *number of views* (Views (Q)), has been ranked quite high by all the feature selection methods. The distribution of the *number of views* in Figure 14 show that

complex questions have less views than the easier ones. This is perhaps not surprising since intuitively easier questions are likely to attract more answerers than complex questions.

In general answer features are ranked quite low. Nevertheless, *answers' score* is ranked among the top 10 positions for both IGR and CFS . The distribution of *answers' score* (Score (A)) in Figure 14 shows that the answers of complex questions receive less points compared to those of easier questions. One possible reason is that complex questions attract fewer more specialised or advanced users, and hence the number of views and votes is subsequently less than for other, less complex, questions. *Question value* (Value (Q)) appears in the top 10 features using either IGR or CFS. It shows that questions with low value are more likely to be complex. As highlighted by Pal and Konstan[216], questions of low value are typically selected by experts. As a consequence, complex questions are more likely to attract expert users. *Asker's community age* appears in the top 10 positions in all the rankings. This shows that the duration of an asker being engaged with a community is indeed a good indicator of question complexity.

[216] *Pal and Konstan (2010)*

Table 20: Top features ranked using their average rank computed from Information Gain Ratio, Correlation Feature Selection and Features Drop for the SERVER FAULT dataset. Type of feature is indicated by *UQ*, *UA*, *Q* and *A* for *Asker*, *Answerers*, *Question* nd *Answers*.

	Info. Gain Ratio			CFS			Feature Drop.		
R.	AR.	*IGR*	Feature	AR.	*AC*	Feature	R.	AF_1	Feature
1	3.4	0.041	*Q. Topics Ent. (UQ)*	1.6	0.050	*Q. Topics. Ent. (UQ)*	1	0.562	*Q. Topics. Ent. (UQ)*
2	6.8	0.030	*Value (Q)*	6.2	0.052	*A. Success Ratio (UQ)*	2	0.583	*A. Success Ratio (UQ)*
3	7.6	0.038	*Score (A)*	6.7	0.068	*Age (UQ)*	3	0.595	*Q. Succ. R. Mean (UA)*
4	9.6	0.034	*A. Success Ratio (UQ)*	7.6	0.074	*Value (Q)*	4	0.598	*Flesch (Q)*
5	10.7	$< .001$	*Q. Mean (UA)*	8.3	0.060	*Views (Q)*	5	0.598	*Age Diff. (UQ)*
6	10.8	0.034	*Views (Q)*	9.1	0.031	*Age Diff. (UQ)*	6	0.598	*Views (Q)*
7	11.0	$< .001$	*Q. Std. Dev. (UA)*	10.1	0.028	*Z-Score (UQ)*	7	0.598	*Q. Rate (UQ)*
8	11.5	$< .001$	*Reputation Dev. (UA)*	10.7	0.034	*Score (A)*	8	0.600	*A. Rate (UQ)*
9	12.5	0.025	*Age (UQ)*	11.1	0.025	*Reputation (UQ)*	9	0.601	*Q. Ans. Mean Dev. (UQ)*
10	12.9	$< .001$	*Q. Rate Mean (UA)*	12.1	0.021	*Questions (UQ)*	10	0.603	*Age (UQ)*

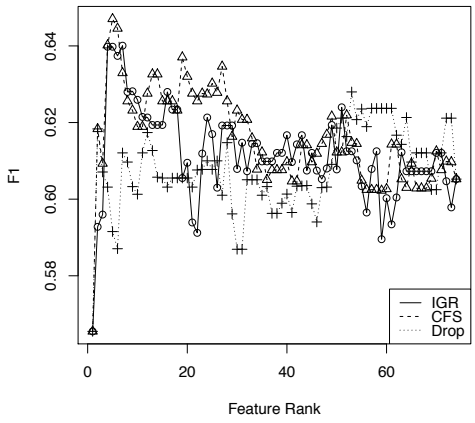


Figure 13: F_1 Vs. feature rank for the Information Gain Ratio, Correlation Feature Selection and Features Drop feature selection methods for the SERVER FAULT dataset.

BEST MODEL SELECTION: A logistic regression model is trained for each set of features selected by different feature selection method. In order to select the minimal and most effective number of features, for each model, features are gradually added according to their discriminative power and the best number of features based on the F_1 score is determined by performing a cut-off when F_1 is the highest.

Using the previous cut-off method, the optimum number of features is 7 using IGR, 5 with CFS, and 53 using the feature drop method (Figure 13).

The question complexity prediction results using the logistic regression model trained on features selected by different feature selection methods are shown in Table 19. It can be observed that all the models perform better than the model that uses all the features. The best result is obtained using CFS ($F_1 = 0.647$) when only 5 features are used. It outperforms the next best result by nearly 4% in F_1. As can be observed from Table 20, among the top 5 features selected by CFS, the top 3 features are related to askers and only the fourth and the fifth features are about the question itself. Hence, it

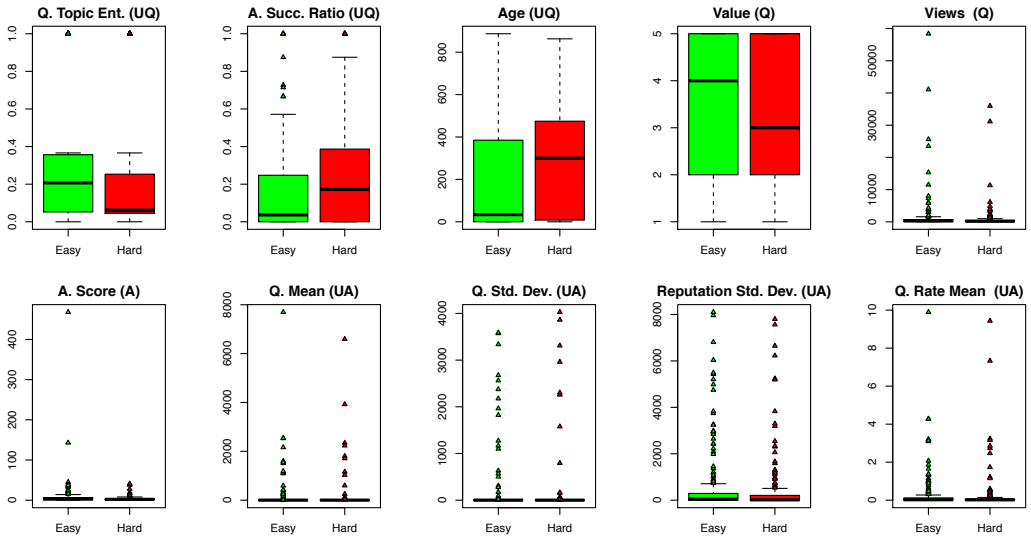


Figure 14: Box Plots representing the distribution of different features and question complexity for the SERVER FAULT dataset. The top row represents the top five features using Correlation Feature Selection. The bottom row shows the top features using Information Gain Ratio (duplicates from the first row are removed).

appears that, for the features used in this chapter, question complexity is primarily determined by asker features. Other types of features only play a marginal role.

6.5 MEASURING COMMUNITY AND USER MATURITY

As communities develop over time, the types of questions asked also evolve accordingly. Identifying the changes in question complexity can be used to understand if community users become more knowledgeable over time. Such measures can be potentially used for measuring the level of knowledge of users (RQ1.3).

6.5.1 Experimental Setting

Intuitively, community maturity can be interpreted as the proportion of complex questions of a community at a given time. Although, maturity can be calculated at different granularity levels, in this thesis, maturity is measured on a monthly basis. At a given time t, given a set of asked questions Q_t, the number of complex questions $|Q_t^{\text{complex}}|$, the community maturity $M(Q_t)$ can be calculated by:

$$M(Q_t) = \frac{|Q_t^{\text{complex}}|}{|Q_t|} \tag{23}$$

With the community maturity measure, the following three tasks are performed. In the first task, the relation between user maturity and user reputation is explored using a $t-$test in order to evaluate if user maturity is a good measure of user knowledge (RQ1.3). In the second task, the evolution of community maturity versus users' community ages is analysed in order to better understand the relation between maturity and user experience. Five different groups are derived from the community users depending on how many days they have been actively engaged with the community. On average, users are engaged 97.78 days. The thresholds are set to $(>1, >10, >20, >50, >100)$ which represent that users are engaged with the community for more than one day, more than 10 days, more than 20 days, etc. In the second task, the questions containing the most discussed topics are extracted by examining the *tags* associated with them. Then, the evolution of community maturity versus topics is measured and discussed to find out if different topics exhibit different maturity evolution curves. For calculating such topic maturity, the proportion of complex questions within a

topic is used. Users who are new to the community (with the community for less than a day) are excluded in order to avoid possible bias incurred by completely new users. By taking users that have more than one day of activity ensures that they contributed more than once.

Although this chapter considers maturity from the point of view of question askers, the maturity measure can be equally used for determining the maturity of answerers by computing it based on the proportion of complex questions answered by a given user. In chapter 8, the answerer maturity is computed instead as the goal is to better identify *best answers*.

6.5.2 *User Reputation and Maturity*

Before studying the evolution of maturity in SF, the relation between reputation and maturity is studied. As previously stated, the aim of the maturity measure is to be an alternative measure of user knowledge as it was hypothesised that *"Knowledgeable users are more likely to answer or ask complex questions"* (H1.3) so that maturity can be used for measuring knowledgeable users (RQ1.3). In this chapter, it is considered that user reputation is similar to user knowledge. Therefore, comparing reputation is similar to comparing user knowledge.

In order to evaluate the previous hypothesis (H1.3), users from SF are separated in two groups depending of their maturity: 1) users that have an overall maturity < 0.5 (24718 out of 33285 user, 74%), and; 2) users that have an overall maturity ≥ 0.5 (8567 out of 33285 user, 26%). Then, a tailed t-test is performed in order to compare

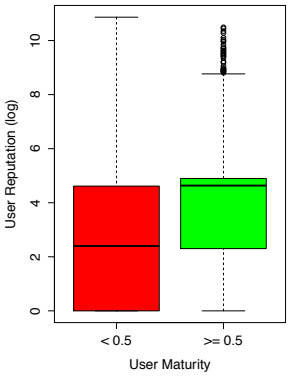


Figure 15: Box Plot representing the distribution of user reputation given different user maturity thresholds for the SF dataset.

both sets and see if there is a difference in reputation between users given their maturity. The two maturity distributions are displayed in Figure 15. As it can be observed it looks like more mature users are likely to have higher reputation which is somehow expected as reputation accumulates over time. The $t-$tests results are described below:

MATURITY DEPENDENCY ($H_1 : \mu \neq \mu_0$): There is a highly significant relation between reputation and maturity with a p-value of 1.510×10^{-22}. This result shows that *user maturity can be used as a proxy measure of knowledge*.

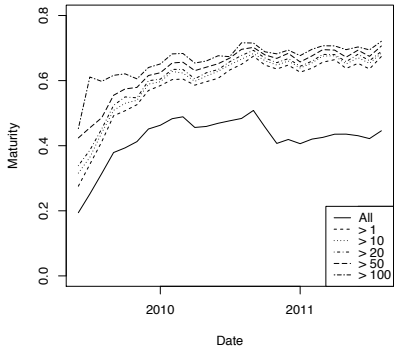
HIGH MATURITY IS ASSOCIATED WITH LOW REPUTATION ($H_1 : \mu < \mu_0$): The result shows a low p-value of 1. Therefore, *a high user maturity is not associated with knowledgeable users*.

HIGH MATURITY IS ASSOCIATED WITH HIGH REPUTATION ($H_1 : \mu > \mu_0$): This result shows that as expected, high reputation is associated with high maturity with a p-value of 7.552×10^{-23}. This observation shows that it is likely that *a high maturity is similar to a high reputation*. Therefore the hypothesis that *"knowledgeable users are more likely to answer or ask complex questions"* (H1.3) appears to be validated and by extension, maturity seems to be a good measure of user knowledge (RQ1.3).

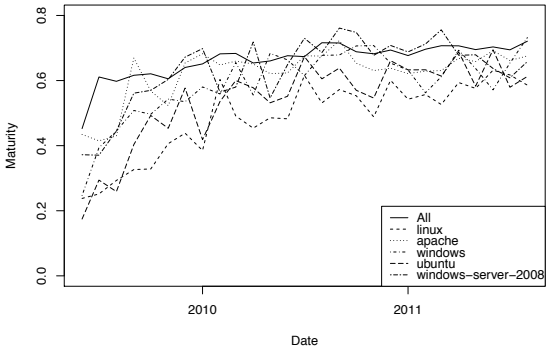
6.5.3 *Community Maturity Evolution*

The evolution of the maturity of the SF community is presented in Figure 16 using the proportion of complex answers contributed over time for the user groups presented in the previous section. A general trend can be observed that regardless of user experience, maturity starts at a low level and then increases until reaching a plateau. This observation can be due to the increase of each user groups over time as the community grows. From the figure it can be deduced that the community age of contributors affect the proportion of complex questions. Unsurprisingly, the longer an asker is with the community, the more complex the questions she asks. However, for all the users, a drop in complexity can be observed at the end of 2010. Such a phenomenon can be explained by a sharp increase of questions asked by relatively new users during this period (the average community age of askers drops from 229 days to 185 days in December 2010). The drop in complexity is also less obvious for users with more than 100 days' engagement with the community which further confirms that the maturity drop is mainly due to the questions posted by relatively new users.

The evolution of users with more than one day's experience with the community is similar with older users representing a more mature portion of the community. The maturity of the whole community seems to stabilise around $M = 0.4$ (see the curve of "*All*" in Figure 16). However, for users that have been present in the community for some time (> 100), the maturity stabilise sensibly higher than the one observed for the whole community with an average maturity of $M \approx 0.64$. In addition, contrary to the community containing all the



(a)



(b)

Figure 16: Monthly community maturity for: Different users (top), and, the most discussed topics for users that have been in the community for more than one day (bottom) for the SERVER FAULT dataset.

users, more experienced users keep increasing over time meaning that the committed contributors are actually maturing. This result shows that the SF community is increasing its skills.

6.5.4 *Topic Maturity Evolution*

This section studies how community maturity evolves with different topics. The focus is on the five most popular tags of the SF community (i.e. the most questions asked with a given tag): *linux*, *apache*, *windows*, *ubuntu* and *windows-server-2008*. Because the interest is in the evolution of the engaged community users, the analysis is constrained on users that have been in the community for at least a day before posting questions ($M_{a>0}$). The maturity evolution of different topics is depicted in Figure 16.

All maturity curves follow a similar pattern that they increase over time although with some oscillations. Different topics exhibit different growth rate in maturity. For example, *linux* shows a slow but sustained maturity increase over time. *Windows-server-2008* seems sensibly more mature than the others at the beginning but its maturity starts to decline at the end of 2010. This can be partly explained by a migration of mature users to a different topic (e.g., *windows-server-2008-r2*) or simply by the increase of new users who tend to post less complex questions. A more detailed explanation of such behaviour is provided in the discussion section (Section 6.7).

6.6 Measuring Complexity and Maturity using Omega

The main drawback of the previous complexity model is that it relies on manual annotations in order to be trained properly. As a consequence, in order to apply the same model on other datasets, time consuming annotations would be needed. Such annotations may be hard to obtain due to the lack of annotators.

In this thesis, two additional datasets are studied, and therefore using the complexity and maturity metrics would require in principle the need to perform manual annotations on each dataset. Unfortunately, additional manual annotation is not a possibility due to the lack of access of experts for the additional datasets.

For dealing with this issue a complexity metric, omega (Ω), is created based on the top 5 features that are associated the most with

question complexity in SF when CFS was used (Section). Although the results may not be directly transferable to other datasets, the studied communities are similar in purpose and structure and hence the metric could be applicable to the other datasets used in this thesis.

6.6.1 *The Omega Complexity Metric*

As previously noted, the omega metric is based on the top five features obtained while using CFS. These features are: 1) asker's question topical focus; 2) asker's ratio of successfully-answered questions; 3) askers' community age; 4) questions' existing value [217], and; 5) questions' views.

[217] *Pal and Konstan (2010)*

A good metric needs to be bounded and deal gracefully with outliers. Accordingly, the values of omega are between 0 and 1 where 0 means an easy question and 1 a complex question. As it can be observed, most of the features that need to be used for creating the complexity metric, are not bounded as a consequence it is needed to transform most values so that they are between 0 and 1. The main issue when normalising values such as the askers' community age or the number of questions' views is that the highest value that can be observed is unknown.

For dealing with such particular case, the $N_{+\infty}(x)$ function is defined so that it is upper bounded even if $x \to +\infty$. The $N_{+\infty}(x)$ function is defined as:

$$N_{+\infty}(x) = \frac{1}{\sqrt{1+\log(1+x)}} \tag{24}$$

Using the previous function, the omega metric noted $\Omega(age, nv, foc, succ, val)$ is created where *age* represents the asker community age, *nv* the number of questions' views, foc the asker's question topical focus, *succ* the asker's ratio of successfully-answered questions, and, *val* the questions' existing value. The community age and the number of questions' views are normalised using $N_{+\infty}(x)$ whereas the questions' existing value is simply divided by 5 so that it is normalised. Since both the topical focus and existing value are negatively correlated with complex questions they are substracted rather than summing them with the other features. The resulting omega metric is defined as:

$$\Omega(age, nv, foc, succ, val) = \frac{1}{5} \times (N_{+\infty}(nv) + succ - N_{+\infty}(age) - \frac{val}{5} - foc + 2) \quad (25)$$

6.6.2 *Omega Vs. Logistic Regression Complexity Model*

Although the quality of the metric is not validated for the SCN forums and CO due to the lack of native annotations, the ability of the metric to predict complex questions on the SF dataset is evaluated by comparing the metric results with the complexity annotations obtained in section 6.4.2. For classifying questions as complex or non complex the arbitrary threshold of 0.5 is applied. Future work

should investigate if this threshold value can be determined automatically. Given the 0.5 threshold value, complex question are considered to be associated with $\Omega > 0.5$ otherwise, if $\Omega \leq 0.5$, a question is annotated as not complex since an omega value of 0 correspond to easy questions while a value of 1 is associated with hard questions.

Similarly to the experiment described in section 6.4.4, the precision (P), recall (R) and the F-measure (F_1) measures are reported for the omega question complexity predictions. The results are displayed in Table 19.

Compared to the best result obtained using CFS, it can be observed that omega produce an higher F_1 with 0.654, but with a lower precision 0.571 instead of 0.648 but much higher recall 0.823 instead of 0.647. Nevertheless, omega may be used with relative success instead of the CFS model. In particular, since omega is a metric, it can be used independently of the availability of dataset annotations.

6.7 DISCUSSION

Automatic identification of complex questions is a hard task. Nevertheless, the concept of question complexity was formalised and it was found that question complexity depends mostly on asker features. In particular, their topical focus, ability to get correct answers and their community age correlate most strongly with complex questions. Users with narrower topical focus are more likely to ask complex questions (Figure 14). This is perhaps not surprising since users who focus on a limited set of topics are more likely to

become experts on those topics and therefore are able to ask more complex questions.

These results complement findings by Pal and Konstan[218] on the question selection behaviour of experts. While they identified that experts prefer questions with low *existing value*[219], it appears that experts tend to ask complex questions which are typically associated with low *existing value*. Such results show that experts not only select questions that are left out by non-experts but also answer complex questions.

[218] *Pal and Konstan (2010)*

[219] *Pal and Konstan (2010)*

As expected, it appears that knowledgeable users are associated with complex questions as high maturity is associated with high reputation (H1.3). This observation confirms the ability of maturity to represent knowledgeable users (RQ1.3)

The approach for measuring maturity is most useful when applied to users that have been involved in the community for more than one day. Such results are coherent with the importance of measuring long-term users rather than uncommitted users. Indeed, committed users are more likely to ask more complex questions and are the ones that provide long-term value to their community. In these circumstances, community managers are more interested in such users compared to users that perform one-off contributions.

Following such observation, it appears that SF is a successful community with long-term askers maturing over time and a maturity above 60%. The analysis of top topics showed that although generally these topics follow a similar maturity evolution trend, they behave somewhat differently. Some topics show high maturity from the start (e.g *windows-server-2008*) while others show slow maturation rate over time (e.g *linux*). Such results may be useful in

different contexts. For instance, topics with high maturity can be used for recognising valuable users while topics with slow but constant maturing rate may help in finding communities and users that show good learning abilities.

The proposed omega metric showed that the complexity of questions can be identified favourably compared to the best model obtained using logistic regression with the advantage of being applicable to non annotated datasets even though its precision is relatively low compared to learned metrics. However, the omega measure benefits from a high recall. Future work should investigate if the omega metric can be improved. In particular, the usage of a sigmoid function could be preferred to the current $N_{+\infty}(x)$ function as it has a more gentle slope compared to the current function. Another line of investigation could be the improvement of the cuttof value for deciding when a question is complex or not complex. Finally, another possibility would be to consider that each individual variable contribute differently to the metric rather than assuming that each parameter is equally important. Nevertheless, in the context of this thesis, it means that complexity and maturity based features can be used in all the datasets thanks to the omega metric.

6.8 SUMMARY

In this chapter, two measures useful for identifying the evolution process of online enquiry communities were presented: question complexity, a measure of the level of expertise required for answering questions; and community maturity, a measure of community knowledge and specialisation. A logistic regression model

was trained for identifying complex questions based on a manually annotated dataset of question pairs extracted from the SF community. Although the modest accuracy of 65% was achieved, it was found that complex questions depends on five key factors: 1) asker's question topical focus; 2) asker's ratio of successfully-answered questions; 3) askers' community age; 4) questions' existing value [220], and; 5) questions' views.

[220] *Pal and Konstan (2010)*

The maturity of the SF community was also measured. This measure established that the community of active users matures over time. In addition, although topics follow a general upward trend, they exhibit different individual maturation processes. Some show high maturity at the beginning while others show a slow maturity rate. Finally, the relation between maturity and user knowledge was confirmed through hypothesis testing (H1.3) meaning that user maturity can be used as a proxy measure of knowledge (RQ1.3). This finding shows that question complexity and user maturity can be used as part of the features selected by the qualitative design methodology introduced in this thesis (RQ1.2).

Since in this thesis additional datasets are used, the omega complexity metric was designed based on the five previous key factors and obtained usable results compared to the regression models. This finding allows to compute the complexity and maturity on the SCN forums and CO datasets. Such features are used at the end of this thesis (Chapter 8) when evaluating the suitability of qualitative design for improving the prediction of *best answers*.

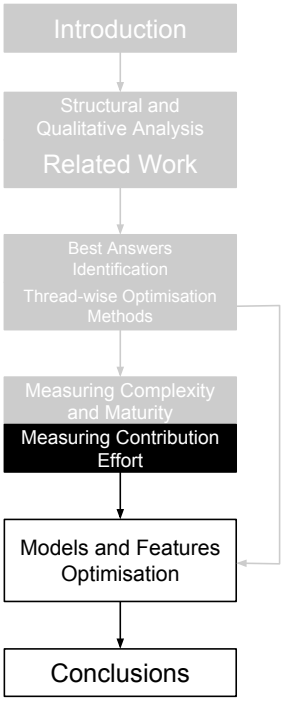
In the following chapter (Chapter 7), the design of more advanced features discovered by the user surveys (Chapter 2) is studied by

introducing contribution effort measures. Such measures may be used as proxy measure of user reactivity.

7

MEASURING CONTRIBUTION EFFORT

Following on the previous chapter on user maturity, the focus is now on designing a measure that takes into account answerer expertise and the ability of answerers to provide quick answers (i.e. user reactivity). The design of such a type of feature is motivated by the perceived relation by community users between contributor expertise, timelessness and quality of answers (Chapter 3).



As part of the qualitative design methodology (RQ1.2), the concept of *contribution effort*, the amount of work a user puts into her answers, is identified as a potential measure of user reactivity (RQ1.4). Therefore, in this chapter, an effort metric that determines on a scale of 1 to 9 the contribution effort associated with a given answer is proposed. The proposed approach is based on the hypothesis that "user reactivity can be estimated from the amount of effort required for generating the words that form an answer" (H1.4).

A set of different models that measure the amount of work each user required in order to produce a given answer is proposed. The proposed models are based on topic models (Chapter 3) and the

evolution of user vocabulary over time. The results obtained in this chapter are reused later on for our attempt to improve *best answer* identification (Chapter 9) in order to determine if qualitative design improves the identification of *best answers* (RQ1.2).

Based on different t-test, the proposed effort model appears to model correctly the effort that users put into contributions. In particular, contribution effort appears to be correlated with fast answers thus validating the hypothesis that associates user reactivity and effort based on word-effort relations (H1.4).

This chapter is divided in 7 parts. First, context where determining the effort of contributions is useful is discussed. Then, the JOINT EFFORT TOPIC MODEL (JET) and AUTHOR JOINT EFFORT TOPIC MODEL (αJET) models are presented. In the third section, the models are validated by comparing them with expected effort behaviour before comparing the perplexity of the models with commonly used topic models. Following the previous results, a lightweight effort evolution analysis on two of the three datasets: SF and CO is performed by discussing the evolution of aggregated effort and different topics. Finally, a discussion of the results is performed before concluding the chapter.

7.1 INTRODUCTION

As part of the qualitative methodology investigated in this thesis (RQ1.2), the user study performed in chapter 2 identified the importance of modelling user reactivity in order to improve *best answer* identification. Rather than modelling such measure directly,

it is proposed to investigate the concept of contribution effort as it can be used for measuring the implicit amount of time a user put into her contributions (RQ1.4).

Much research has been devoted to understanding how users behave in online communities through, for example expertise identification, churn prediction and content understanding. Although existing research allows community managers to understand the status of the communities they administrate, other factors such as the topical effort required by community members to contribute has not been systematically studied. Understanding the effort or the amount of work and time that each user requires in her contribution could help in identifying *low effort* topics as well as relatively *high effort* topics. It is also useful to detect user-level effort patterns, for example, users who reduce their contribution efforts could signal a loss of interest in the community. Such knowledge would allow community managers to act upon emergent events such as churn, dying topics and low contributions. In the context of the automatic identification of *best answers* investigated in this thesis, measuring effort may be particularly useful as expertise and user reactivity are expected to correlate with good content (Chapter 2).

In general, it is difficult to effectively measure users' contribution effort. A possibility is that users tend to *contribute uniformly over a short period of time* and that a deviation in their contribution patterns indicates a change in the amount of work or time allocated to each of their contributions. We also assume that each user tends to *use the same set of vocabulary terms and any variations on vocabulary terms could be used as a proxy measure of effort* (i.e. atypical vocabulary yields more effort than terms commonly used by

the user) (H1.4). The main benefit of such an assumption is that it does not rely on non-standard community features found in domain specific communities (e.g. community ratings, user reputation, network structure). To this end, the presented approach can be applied to a large range of communities as long as authors and timestamped textual content is available such as the datasets used in this thesis.

Following these considerations, two Bayesian models that capture the effort required by users to contribute to different topics are proposed. The models are based on different topic models such as the LDA and JST that were discussed in chapter 3.The evolution of effort patterns on two of the datasets studied in this thesis, CO and SF, are also studied. The contributions of this chapter are:

1. Introduce the concept of contribution effort, a value representing the level of labour and time required for contributing or posting to a community.
2. Based on the concept of effort, two measures of effort (STAN and ASTAN) are created by relying on the concept of Stanines, a grading measure used in examination marking schemes based on z-scores.
3. Present the JET model and its authored version, the αJET model designed for balancing out STAN and ASTAN effort modelling weaknesses.
4. Investigate the evolution of community effort in two different communities and demonstrate that contribution effort is influenced by user dynamics.
5. Investigate if user reactivity (i.e. answering speed) can be estimated from the amount of effort required for generating the words that form a given answer (H1.4).

7.2 JOINT EFFORT TOPIC (JET) MODEL

There is no clear definition of contribution effort. Also, there has been no significant prior work on representing and learning effort from online communities (Chapter 3). In this section, a definition of contribution effort and two hypotheses about how it can be derived are proposed. Following the definition and these hypotheses a measure of effort based on the concept of Stanines, a method used in examination grading schemes based on z-scores is provided. Then this chapter introduces two Bayesian models that learn the effort associated with topics and documents based on community and user contribution patterns. Finally, this chapter discusses how to use the effort measured by Stanines to set the word effort priors in the proposed Bayesian models.

7.2.1 Defining Contribution Effort

In this thesis the concept of effort is viewed as the level of labour or time required by an individual or community to perform a given task. Therefore, contribution effort is defined as follow:

Definition 7.1 (Contribution Effort)**.** *Contribution effort is a value representing the amount of labour (or time) required for contributing or posting to a community.*

Although the amount of effort required for performing a contribution may be correlated with the quality of the produced content, the

definition of contribution effort is independent from the contribution quality since the aim here is to simply measure the reactivity and contributing ability of the individuals that form a community independently from the quality of the content produced. In the context of community wide answer value identification, this means that effort is not a direct measure of quality but a measure of contribution ability. For example, in the Q&A communities studied in this thesis, a given question may be hard yet the answering effort is low. For some contributors, their answers may be of low quality even though the effort involved is high. As a summary, *contribution effort is a measure of contribution ability rather than content complexity*.

Measuring contribution effort is a rather complex task since it is generally impossible to account for the time that users have invested in their contributions. Additionally, user effort cannot be annotated directly by a third party since effort is highly dependent on authors' personal ability; meaning that it would be very difficult to create a gold standard of effort can be created without consulting the actual authors of posts. This chapter postulates that the contribution effort of a particular post can be decomposed into word-level effort (H1.4). The intuition is that vocabulary usage carry information about one's ability to employ a given term in different contexts. Accordingly, for a user, preferred vocabulary can be considered easier compared to rare or odd terms. This idea is generalised by considering that preferred vocabulary terms are stable within a certain period of time. Words used more often during this period would

incur lower contribution effort. On the contrary, words used less often would incur higher contribution effort. This leads to the formulation of the following hypothesis:

Hypothesis 6 (Vocabulary Preference)**.** *The effort associated with a given contribution is correlated with the preference associated with particular vocabulary terms. The more a given term is contributed by a user or community the less contribution effort it involves. On the contrary, the less a term is contributed the more the contribution effort incurred.*

Users' average contribution effort within a certain period of time is also considered to remain relatively stable. For example, a user who is a *Linux expert* tends to answer more Linux-related questions. Her contribution effort will more or less remain similar within a certain timespan. Any deviation from this stable pattern will indicate a change in contribution effort. The above can be summarised in the following hypothesis:

Hypothesis 7 (Effort Pattern Stability)**.** *Within a certain period of time, the average amount of labour provided remains stable and is independent from the total number of vocabulary terms used during the given time period.*

Although other information such as user expertise or community network structure can be potentially integrated into user effort estimation, these features are left out in order to keep the approach

simple and applicable to any type of community that has author information and timestamped textual data.

7.2.2 *Measuring Effort with Stanines*

As previously highlighted, changes in contribution effort are detected by measuring the deviation from the stable contribution effort pattern. A deviation to the left means more effort than usual while a deviation to the right means less effort than usual. The standard score or z-score can be used to measure the number of standard deviations above or below the contribution effort pattern. While z-scores give real numbers, it is important to discretise them so that they can be used to set priors of the proposed JET model which will be described in the next subsection.

Although different methods exist for discretising z-scores such as Stanine (STAndard NINE) and Sten, the proposed model uses Stanine. In context of effort measurement, a Stanine of one means costly contributions, five average effort and nine the least effort. Sten is similar to Stanine except that is has 10 possible values. Using Stanine with the odd number scale makes it easier to identify the centre of the effort distribution of a community (i.e. where contribution is neither costly nor effortless).

The relation between z-scores and Stanines is straightforward. The Stanine of a given raw value x from a given a population X with

mean μ and standard deviation σ is given by first calculating the z-score, z associated with x and then applying Equation 26:

$$Stanine(z) = \begin{cases} 1 & \text{if } z < -1.75 \\ 2 & \text{if } -1.75 \leq z < -1.25 \\ 3 & \text{if } -1.25 \leq z < -0.75 \\ 4 & \text{if } -0.75 \leq z < -0.25 \\ 5 & \text{if } -0.25 \leq z < 0.25 \\ 6 & \text{if } 0.25 \leq z < 0.75 \\ 7 & \text{if } 0.75 \leq z < 1.25 \\ 8 & \text{if } 1.25 \leq z < 1.75 \\ 9 & \text{if } z \geq 1.75 \end{cases} , z = \frac{x-\mu}{\sigma} \quad (26)$$

Equation 26 is derived from the normal distribution $\mathcal{N}(\mu, \sigma)$. As displayed in Figure 17, the normal distribution is divided into nine equal parts except for the first part and the last part (Stanine 1 and 9) and each part is given a Stanine. Then, it is simply required to calculate the z-scores of a raw value, and associate the corresponding Stanine using $\mathcal{N}(\mu, \sigma)$. The normalisation method used by the Stanine approach makes sure that unusually high values lead to high scores whereas unusual low ones lead to the low scores. Similarly, a small deviation from the population mean means an average effort. This approach makes Stanines particularly suitable for effort modelling since the focus is on modelling the relative amount of work users put into their contributions. Therefore, most of user

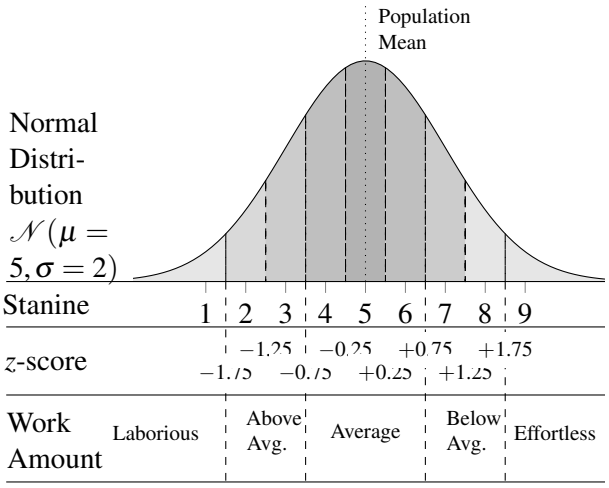


Figure 17: Relations between z-scores, stanines and work amount (contribution effort).

contributions should lead to average effort while outstanding and unusual contributions would result in high or low effort.

Following the two previous hypotheses, the effort of a given document is calculated using the Stanine approach based on the distribution of word counts within the current time period and the distribution of the word counts during the last M time periods so that each word can be associated with an effort value ranging from 1 to 9. The effort of a given word can be calculated either for a full community or for each individual. For a given user, the effort $e_{p,w}$ associated with a given word w in period p can be calculated based on the deviation from the number of times the same word has been observed in the last M periods and the current period for the same user. First, the z-score is calculated using the previous M periods word distribution w_M, the current period p and the number of word occurrences w_p. Second, Equation 26 is applied for obtaining the effort of w. The effort of a document is then derived using the document-effort distribution created by averaging the word-effort distributions of each word contained in the document. In this thesis, the general effort measure is refereed as STAN and the author-specific effort measure as ASTAN.

7.2.3 *The Joint Effort Topic Models (JET/αJET)*

The previous subsection discussed how effort associated with each document can be combined based on the effort value measured for each term occuring in the document using the concept of Stanines and term distributions in the form of STAN or ASTAN. Such a coarse measurement does not account for cases where users can choose different words to convey similar semantic meanings. Also, STAN or ASTAN based on historical word usage patterns cannot handle new words and gives unreliable effort estimation to words occurring relatively rare in the past. As such, it makes sense to group words bearing similar semantics into latent topics and estimate effort at the topic level rather than the word level. The overall effort of a document can then be modelled as a mixture of topic-level efforts and the word-level effort of recently introduced terms can be estimated more reliably by using their topic-level effort.

A Bayesian model, which jointly models topics and topic-level efforts based on LDA ,[221] is proposed. As presented in chapter 3, LDA (Figure 18) is a well-known admixture model for generating hidden topics given textual documents. In LDA, each document is a mixture of topics and each topic is a probability distribution of words. The generative procedure is generally a two steps process where: 1) For each document d, a multinomial distribution ϕ' over topics is randomly sampled from a Dirichlet with parameter β', and; 2) For each word position in document d, a topic k is randomly sampled from the topic distribution ϕ'_d, and a word w is generated by randomly sampling from a topic-specific multinomial distribution ψ'_k (Figure 18).

[221] *Blei et al. (2003)*

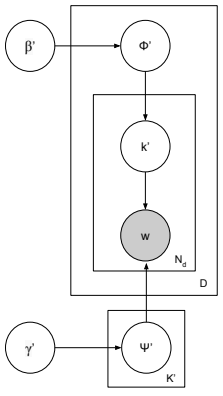


Figure 18: The Latent Dirichlet Allocation (LDA).

To additionally model topic-level effort, the previously proposed JST[222] model can be used (Figure 19). The model was originally applied to topic-associated sentiment detection from text. It assumes that a document is modelled with a mixture of sentiment labels (positive, negative or neutral), under which a set of topics is associated with each of the sentiment labels, and words are associated with both sentiment labels and topics. To generate each word, a sentiment label e is sampled from the per-document sentiment distribution σ_d, then a topic k is sampled from a sentiment-specific topic distribution $\phi_{d,e}$, and finally, a word w is generated from a per-corpus word distribution $\psi_{e,k}$ conditioned on both sentiment label e and topic k. If the sentiment distribution is replaced with an effort distribution, the JST model can be used to model topic-level effort as the fundamental idea of the proposed effort model is to associate effort label to individual words and then effort distributions to topics. In JST, the sentiment prior knowledge that encodes words typically bearing positive or negative polarities comes from a fixed sentiment lexicon. In this thesis, the word-level effort prior is dynamically computed for each time period based on the aforementioned Stanine method. In addition, author information needs to be incorporated in order to enable the measure of author-specific effort.

[222] *Lin and He (2009)*

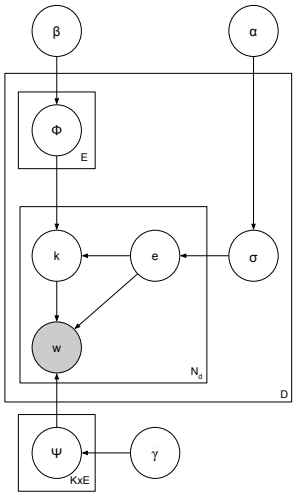


Figure 19: The Joint Sentiment Topic (JST) model.

With the JST model added with the period plate outside the document plate and the author plate outside the period plate, author-specific topic effort can be learned for different time periods. However, the topics learned under the effort labels for different author are not directly comparable. In order to cross compare effort required for the same topic by different authors at the same time period or the same author at different time periods, topics need to be

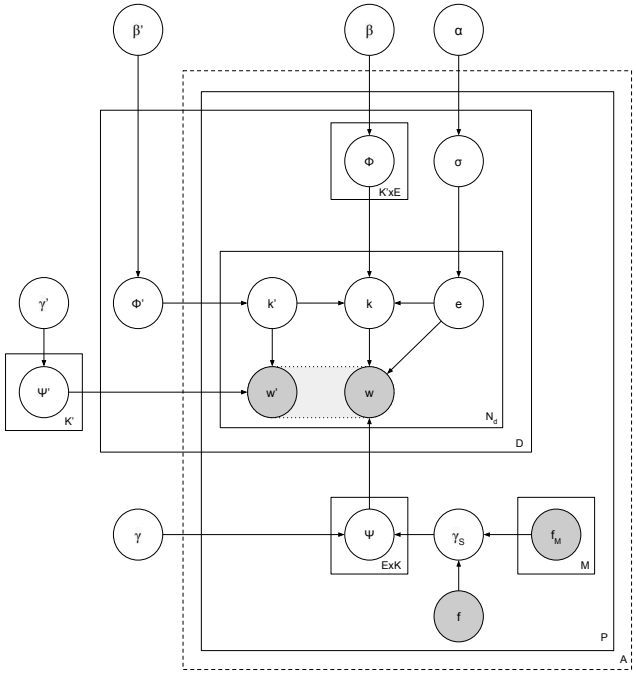


Figure 20: The Joint Effort Topic (JET) model (without dashed plate) and Author Joint Effort Topic (αJET) model (with dashed plate).

generated from documents independent from period and author first and then they need to be linked to period (or/and author) specific effort-topics. For this reason, the JET model (without the author plate) and the αJET model (with the author plate) are proposed as illustrated in Figure 20.

In the proposed JET orαJET models, instead of generating a word w for each word position in a document, a pair of words (w, w') is generated simultaneously under the constraint $w' = w$. Essentially, word w is duplicated at each word position. The word w' is generated by a standard LDA model while w is generated by a JST-like model except that the topic k associated to w is sampled conditional on both effort e and the topic k' which has been previously drawn from LDA and is independent from both periods and authors. The primary topic is noted k' while the effort-topic is noted and k.

Consider a set of A users $\mathscr{A} = \{a_1, a_2, \ldots, a_A\}$ and that for each user a, a set of non overlapping P time periods $\mathscr{P} = (p_1, p_2, \ldots, p_P)$.

Each time period can be for example a monthly period or a daily period. For each user and a time period p in $\mathscr{P}$, a sequence of D documents $\mathscr{D} = (d_1, d_2, \ldots, d_D)$ that have been contributed during the time period p is defined. Assume that each contributed document d is composed of a sequence of N_d indexed terms from a vocabulary of size V, $\{w_1, w_2, \ldots, w_{N_d}\}$. Finally, let K be the total number of primary topics which are independent from periods or authors, K' be the total number of effort topics and $E = 9$ the number of effort labels.

The generative process which corresponds to the JET and αJET models shown in Figure 20 is presented as follows (the elements in **bold** are only valid for αJET):

- For each primary topic $k' \in \{1..K'\}$, draw $\psi'_k \sim Dir(\gamma')$
- For each document d,
  - draw $\phi'_d \sim Dir(\beta')$
  - draw $\sigma_d \sim Dir(\alpha)$
  - For each effort label e and primary topic k', draw $\phi_{d,k',e} \sim Dir(\beta)$.
- For each period $p \in \{1..P\}$ **with author a**
  - For each effort label $e \in \{1..E\}$ and for each effort topic $k \in \{1..K\}$, draw $\psi_{\mathbf{a},p,e,k} \sim Dir(\gamma_s)$
- For each document d in period p
  - For each word position i in document d:
    - choose a primary topic $k'_i \sim \phi'_d$,
    - choose an effort label $e_i \sim \sigma_d$,
    - choose an effort-topic k_i conditional on both primary topic k'_i and effort label e, $k_i \sim \phi_{d,k'_i,e_i}$,

* choose a word w'_i from the distribution over words defined by the topic k'_i, $w'_i \sim \psi'_{k'_i}$.
* choose a word $w_i = w'_i$ from the distribution over words defined by the period p, the topic k_i, the effort label e_i **and author a**, $w_i \sim \psi_{\mathbf{a},p,e_i,k_i}$.

It is worth noting that in the above process, γ_s is the Dirichlet prior of the effort-topic-word distribution which is set using the word-level effort values estimated by Stanines (see Section 7.2.4).

The Gibbs sampler algorithm for JET and αJET will sequentially estimate k'_t, k_t and e_t given a document d from the distributions over each variables given the current value of all other variables and the data. The conditional posterior for k'_t, k_t and e_t is denoted in equation 27 where the N notation represent the counts of each associated variable (the variables in **bold** are only relevant for αJET) assuming $\Lambda = \{\alpha, \beta, \beta', \gamma, \gamma'\}$:

$$P(k'_t = x, k_t = y, e_t = z \mid w', w, k'_{-t}, k_{-t}, e_{-t}, \Lambda) \propto \frac{\{N_{d,x}\}_{-t} + \beta'}{\{N_d\}_{-t} + K'\beta'} \cdot \frac{\{N_{x,w_t}\}_{-t} + \gamma'}{\{N_x\}_{-t} + V\gamma'} \cdot \frac{\{N_{d,e}\}_{-t} + \alpha}{\{N_d\}_{-t} + E\alpha} \cdot \frac{\{N_{d,x,y,z}\}_{-t} + \beta}{\{N_{d,x,z}\}_{-t} + K\beta} \cdot \frac{\{N_{\mathbf{a_t},p_t,y,z,w_t}\}_{-t} + \gamma_s}{\{N_{\mathbf{a_t},p_t,y,z}\}_{-t} + V\gamma_s} \quad (27)$$

7.2.4 *Setting Model Priors*

A key element of generative models is the use of good priors that ensure accurate and meaningful inference. A common approach is to use uniform priors so that the inference task is only influenced by the data used during the learning phase. The hyperparameters, α, β, β' and γ' do not necessarily require biased priors. For each of them,

a uniform prior is used. Nevertheless, applying uniform priors to γ, a Dirichlet prior of effort-topic-word distributions would result in a hierarchical topic model where the effort component would be treated as another topical dimension rather than as an effort dimension making it useless for the effort modelling task.

As discussed in section 7.2.2, STAN and ASTAN can be used as a rough estimation of word-level effort which can be subsequently used to set priors dynamically for JET and αJET. By using this approach, the effort associated with words and documents for a given time period and author can be biased. The bias permits an accurate repartitioning of the effort associated with words.

For each word w_i in document d at time period $p \in \mathscr{P}$ and, in the case of αJET, author $a \in \mathscr{A}$, the occurrence frequency of w_i in p (for author a if αJET), f, is used in conjunction with the distribution of the same word over the M previous time periods, denotes as f_M. Given such information, the Stanine can be calculated using f and the previous period word counts distribution f_M. Then, the returned value can be used as the word-level effort prior for w_i.

Rather than directly using the effort index returned by STAN or ASTAN, the word-level effort prior is smoothed using a normal distribution centred around the effort index. This approach has a couple of benefits. First, it allows a soft assignment of effort label given a word. For example, if the Stanine returned is 5, the probability of word associated with an effort label of 4 or 6 would also be high. Second, the probability of generating any word given an effort label is always positive. In another words, a word has a non-zero probability associated with any one of the effort labels although some of the association probabilities could be low.

There are two special cases that need to be taken care of. First, when $p \leqslant M$, there are no sufficient historical data to calculate the effort index reliably; Second, when the current word has never appeared in the previous M periods, it is not possible to estimate the effort index for unseen words based on STAN or ASTAN. In both cases, a default uniform word prior is used instead of the Stanine based prior and the JET and αJET generative process is used for estimating the true word-level effort of newly observed or recently introduced terms based on their topic-level effort allocation. We use γ to denote the default uniform prior and γ_s the STAN or ASTAN based word prior.

7.3 MODEL EVALUATION THROUGH HYPOTHESES TESTING

Since it was not possible to ask content owners to estimate the effort of their own past contributions and third party effort annotations are not appropriate for evaluating contribution effort, an evaluation method that does not require direct effort labels is required. A solution is to measure the ability of effort models to validate expected behaviours such as correlations between user contributions and the associated effort. In order to show the validity of the proposed model, a set of test hypotheses is defined (Section 7.3.1) about the expected behaviour of the model and each assumption is validated through a paired t-test by comparing effort behaviour over time under the conditions of a given hypothesis (e.g. effort behaviour for the most active contributors against the least active users).

Hypotheses are tested by focusing on CO and SF datasets used in this thesis as they are smaller than the SCN dataset and make effort computation easier. First, the validity of the STAN and ASTAN approaches is evaluated by testing the effort results against each hypotheses. If the statistical significance is high enough, STAN and ASTAN are considered good representations of contribution effort. After testing the Stanine based methods, a similar task is performed on both JET and αJET and the STAN and ASTAN statistical significance is checked. If the statistical significance still holds, the proposed models are considered to be able to capture topic-level contribution effort and to properly estimate the effort of newly observed or rarely occurred terms.

For each of the tested methods, a window size of $M = 4$ months is selected and words appearing with a relative frequency below 10^{-5} or higher than 0.99 in each of the studied dataset are filtered out. For JET and αJET, the following parameters are set: $\alpha = \beta = \beta' = 10^{-4}$ and $\gamma = \gamma' = 10^{-7}$. For the topics $K' = 5$ and $K = 15$ are selected. Although different methods exists for setting the hyperparameters automatically, the chosen priors work well for the proposed experiments. The estimation of the hyperparameters in a more principled way is left as future work.

The experiments are conducted on the CO and SF datasets discussed in chapter 2. As many community users do not contribute much, answers that are contributed by authors that have at least contributed 5 times in the community are selected. For the CO dataset, the retained data is composed of 8,272 (84.24%) answers, 327 (6.62%) users and 4,555 stemmed words. For SF, the final dataset

consists of 140,436 (86.47%) answers, 4,060 (7.85%) users and a total of 3,979 stemmed terms.

7.3.1 Evaluation Hypotheses

For the purpose of model evaluation, a set of three test hypotheses is asserted. Each test hypothesis expects different behaviour for high effort and low effort. As a consequence, it is easy to perform a paired t-test by splitting the documents into two groups, one corresponding to documents involving high effort and the other one low effort. Our test hypotheses are as follow:

Test Hypothesis 1 (Activity Level)**.** *Users who contribute a lot (post more messages) have lower effort than users that contribute less (TH_1).*

To test this hypothesis, the top 10% users with the most contributions and the top 10% users with the least contributions are selected for each dataset. Given the two sets of users, the average monthly effort is calculated for each set and the effort of active users is tested in order to check if it is significantly lower than the least active users over time.

Test Hypothesis 2 (Time to response)**.** *Users take more time to respond on documents that require more effort (TH_2).*

This hypothesis is quite intuitive. It states that given a question, if a user takes more time to supply an answer, then more effort is incurred.

Although the access to the actual time users spent on each of their posts is not available, it can be roughly estimated by using the response time based on the difference between question time and answer time. Whilst not as accurate as using users' click history, the difference between question time and answer time can be used as a lower bound estimate of users' time to response behaviour. As a consequence, if statistical significance is observed then it can be expected a similar or better result for stronger time to response measures. Similarly to the previous hypothesis the top 10% documents with the fastest response time and the top 10% documents with the slowest response time are picked. Given the two sets, the average monthly effort for each set is calculated and it is tested if the effort associated with fast response is significantly lower than that with slow response over time.

Test Hypothesis 3 (Term Preference)**.** *Users have lower effort when using terms they are familiar with (TH_3).*

Users that always use a same set of vocabulary terms are more likely to have lower effort. For each user, the perplexity of generating a document given words they have used in the previous M periods is calculated. A lower perplexity means a higher probability to generate a document. The opposite means a lower probability. Like in the previous cases, the top 10% documents with the lowest perplexity and the top 10% documents with the highest perplexity

are picked. Given the two sets, the average monthly effort for each set is calculated and and it is tested if the effort associated with low perplexity is significantly lower than high perplexity content over time.

7.3.2 *Hypotheses Testing*

Each of the test hypotheses is tested on all of the studied datasets. The expected behaviours are: 1) Activity level (TH_1): The effort of active users should be lower than the effort of less active users ($\mu > \mu_0$) ; 2) Time to response (TH_2): The effort of fast responding users should be lower than the effort of users that respond slowly ($\mu < \mu_0$) , and; 3) Term preference: The effort associated with users that post using familiar terms is lower than those that post using less familiar terms ($\mu < \mu_0$). Each testing result is reported in Table 21.

Since it can be observed in Figures 21 that the behaviour of effort patterns of the SF dataset are unstable during the first ten months ($p < 10$), each SF test hypothesis is also computed on the period subset $p \in [10,31]$ where the effort is more stable. The testing results are presented in Table 22. In the rest of the chapter SF_{10+} denotes the SF dataset that excludes the efforts values computed for the periods $p < 10$. The behaviour instability for the earlier periods can be explained by large differences in questions and answers post rates during the first months of existence of the SF community compared to the later months (the post rate Median Absolute Deviation (MAD) during the first 9 periods is $MAD_{p\in[1,9]} = 6245.18$ while the variance for the further periods is the much lower $MAD_{p\in[10,31]} =$

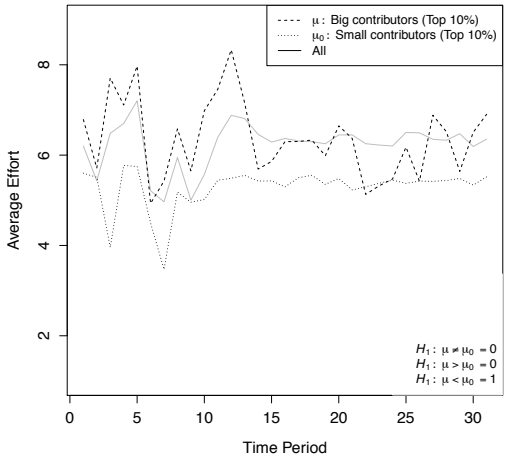
892.84). Such a difference generate noisy effort patterns that can lead to less accurate paired t-tests results.

Table 21: Hypothesis testing using a paired t-test for STAN, ASTAN, JET and αJET for the *Cooking* (CO) and *Server Fault* (SF) datasets. Hypotheses: TH_1: Activity level (expected $TH_1 : \mu > \mu_0$); H_2: Time to response (expected $TH_2 : \mu < \mu_0$), and; TH_3: Term preference (expected $TH_3 : \mu < \mu_0$).

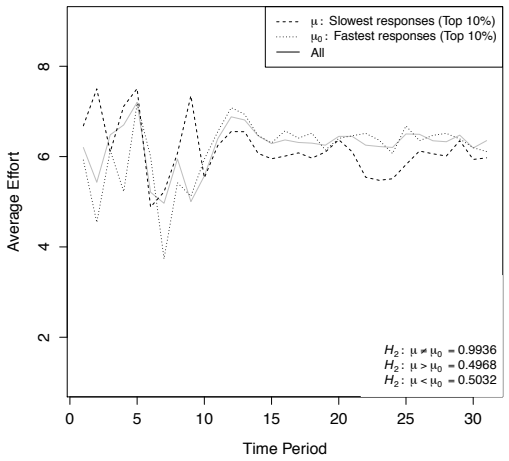
			P-values		
Dataset	Model	TH_n	$TH_n : \mu \neq \mu_0$	$TH_n : \mu > \mu_0$	$TH_n : \mu < \mu_0$
Cooking	STAN	TH_1	0.89	0.445	0.555
		TH_2	0.596	0.702	0.298
		TH_3	0.128	0.06385$^{\cdot}$	0.936
	ASTAN	TH_1	0.07818$^{\cdot}$	0.03909*	0.961
		TH_2	0.0298*	0.985	0.0149*
		TH_3	0.003448**	0.998	0.001724**
	JET	TH_1	0.707	0.354	0.646
		TH_2	0.968	0.516	0.484
		TH_3	0.009257**	0.004629**	0.995
	αJET	TH_1	0.155	0.07738$^{\cdot}$	0.923
		TH_2	0.107	0.946	0.05374$^{\cdot}$
		TH_3	0.001958**	0.999	0.0009789***
SF	STAN	TH_1	0.564	0.718	0.282
		TH_2	0.712	0.644	0.356
		TH_3	0.07902$^{\cdot}$	0.03951*	0.96
	ASTAN	TH_1	$3.4e-08^{***}$	$1.7e-08^{***}$	0.999999
		TH_2	0.148	0.926	0.07403$^{\cdot}$
		TH_3	$2.5e-09^{***}$	0.999999	$1.2e-09^{***}$
	JET	TH_1	0.06595$^{\cdot}$	0.967	0.03297*
		TH_2	0.489	0.756	0.244
		TH_3	0.08999$^{\cdot}$	0.04499*	0.955
	αJET	TH_1	$1.4e-07^{***}$	$7.1e-08^{***}$	0.999999
		TH_2	0.994	0.497	0.503
		TH_3	$2.2e-10^{***}$	0.999999	$1.1e-10^{***}$

Signif. codes: p-value $<$ 0.001 *** 0.01 ** 0.05 * 0.1 .

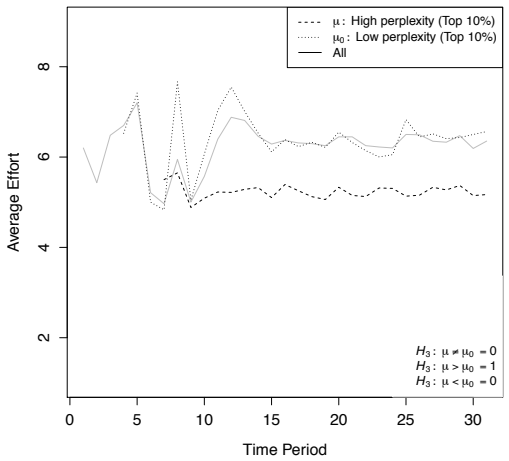
Activity Level In general, STAN does not show any significance whereas ASTAN shows that it is more likely that users who are frequent contributors have lower effort than rare contributors (p-value < 0.05 for CO, p-value $\ll 0.001$ and for SF and SF_{10+}, p-value $\ll 0.001$). It is perhaps not surprising that STAN does not show any significance compared to ASTAN since the effort of users is averaged out over all documents by all users in STAN leading to rather inaccurate effort calculation (i.e. efforts tend to become neutral at the community level). Unsurprisingly, JET does not generate any



(a) Activity Level



(b) Time to Response



(c) Term Preference (Perplexity)

Figure 21: αJET hypotheses tests for the *Server Fault* (SF) dataset. Hypotheses: TH_1: Activity level (expected $TH_1 : \mu > \mu_0$) (a); TH_2: Time to response (expected $TH_2 : \mu < \mu_0$) (b), and; TH_3: Term preference (expected $TH_3 : \mu < \mu_0$) (c).

Table 22: Hypothesis testing using a paired t-test for STAN, ASTAN, JET and αJET for the *Server Fault* (SF) dataset for time periods $p \in [10,31]$. Hypotheses: TH_1: Activity level (expected $TH_1 : \mu > \mu_0$); H_2: Time to response (expected $TH_2 : \mu < \mu_0$), and; TH_3: Term preference (expected $TH_3 : \mu < \mu_0$).

		P-values		
Model	Hypotheses (TH_n)	$TH_n : \mu \neq \mu_0$	$TH_n : \mu > \mu_0$	$H_n : \mu < \mu_0$
STAN	TH_1: Contributions	0.037^{*}	0.981	0.019^{*}
	TH_2: Time	0.004^{**}	0.998	0.002^{**}
	TH_3: Perplexity	$5.4e-05^{***}$	$2.7e-05^{***}$	0.999999
ASTAN	TH_1: Contributions	$1.7e-06^{***}$	$8.3e-07^{***}$	0.999999
	TH_2: Time	$9.8e-07^{***}$	0.999999	$4.9e-07^{***}$
	TH_3: Perplexity	$7.0e-13^{***}$	0.999999	$3.5e-13^{***}$
JET	TH_1: Contributions	$0.057^{\cdot}$	0.971	0.028^{*}
	TH_2: Time	0.008^{**}	0.996	0.004^{**}
	TH_3: Perplexity	0.0001504^{***}	$7.5e-05^{***}$	0.999999
αJET	TH_1: Contributions	$1.2e-05^{***}$	$5.9e-06^{***}$	0.999999
	TH_2: Time	$2.2e-07^{***}$	0.999999	$1.1e-07^{***}$
	TH_3: Perplexity	$4.5e-13^{***}$	0.999999	$2.3e-13^{***}$

Signif. codes: p-value $<$ 0.001 *** 0.01 ** 0.05 * 0.1 .

significance for CO and in the case of SF and SF_{10+}, the statistical significance is reversed ($\mu < \mu_0$). Again, these results can be attributed mainly as the effect of user aggregation and the use of STAN priors. On the contrary, for αJET, the expected behaviour tends to be largely confirmed (p-value $\ll 0.001$) for SF and SF_{10+}.As a result, the observations of ASTAN still hold for αJET and therefore αJET learns effort from the biased prior.

Time to Response The significance results generated using ASTAN somewhat match the expected behaviour with some level of significance observed for both datasets. However, it is worth noting that the score for SF is relatively low (p-value $= 0.07403$). However, for SF_{10+}, high significance is observed (p-value $= 4.9e^{-07}$). Such a result may show that when the number of contributions of bigger Q&A community (SF $\gg$ CO) becomes stable, responses times become more deterministic of contribution effort as users get split into highly specialised users and new users. No significance is observed for JET. αJET gives a nearly significant result for CO but not for SF (p-value $= 0.503$). However for SF_{10+}, the results

are significant (p-value $= 1.1e^{-07}$) thanks to a more stable community. Given such results, the observations using ASTAN holds for αJET.

Term Preference For the last hypothesis, STAN is again not very efficient for measuring effort but ASTAN does a relatively good job for measuring effort (p-value < 0.01 for CO and p-value $\ll 0.001$ for SF and SF_{10+}) and matches the expected user behaviour. JET however gives an opposite result that users have higher effort when using familiar terms. This is against this thesis intuition. Nevertheless, αJET generates slightly more significant results compared to ASTAN (p-value $\ll 0.001$). As with the previous cases, the ASTAN assumptions still apply to αJET for the last hypothesis.

As a summary, the Stanine based effort measure is inline with effort expectations. However, only the authored version is accurate in representing effort. The assumptions about effort still holds true for αJET with the prior derived from ASTAN. Finally, stable communities display better effort stability in relation to the test hypotheses. Such result may be explained by the relative user instability of large and young communities such as SF (for $p \in [1,9]$) where a small number of users account for a large and variable number of contributions and have important effort variations between close time periods (Figure 21). As the community grows larger, the variations become smaller and it becomes clear that effort is correlated with activity levels, time to response patterns and terms preference.

Table 23: Perplexity for the *Cooking* (CO) and *Server Fault* (SF) datasets with different number of primary topics (denoted in brackets under the "Model" column) and effort-topics.

		Number of Effort-Topics					
Dataset	Model	2	4	8	16	32	64
Cooking	LDA	1066.2	1013.8	950.9	861	745.6	638.7
	JST	843.3	731	615.6	486.3	368.8	259.1
	JET(1)	480.6	365.8	263.5	178.4	119.8	80.4
	αJET(1)	70.4	62	56	51.1	49	47.7
	JET(5)	107.8	93.7	76	59.1	44	32.3
	αJET(5)	17.3	18	18.8	19.8	20.3	20.7
SF	LDA	741.4	712	706.1	674.3	645.8	608.0
	JST	659.7	632.6	590.9	546.8	492.3	427.5
	JET(1)	484.7	428.9	362.5	290	218.1	155.8
	αJET(1)	56.1	47.1	40.4	36	32.9	31
	JET(5)	103.4	98.5	91	81	68.70	56.21
	αJET(5)	14.5	15.7	16.9	18.1	18.9	19.6

7.4 MEASURING PERPLEXITY

In order to compare the generative power of the proposed models both JET and αJET are compared against LDA and JST by calculating the performance of each model in term of predictive perplexity (Chapter 2).

The perplexity of LDA, unbiased JST (i.e. without effort bias), JET and αJET are calculated for each of datasets. For each dataset, a 90%/10% training/testing split is used and the perplexity is calculated on the test set with an increasing number of effort-topics $K \in \{2,4,8,16,32,64\}$. For JET and αJET, the impact of different number of primary topics $K' \in \{1,5\}$ is also tested.

The perplexity of each model performs as expected and generally decreases with the increase of primary and secondary topics. A detailed look at the individual performance of each model (Table 23) shows that JST outperform LDA and both JET and αJET have a much lower perplexity and hence superior performance than both

LDA and JST. When using only one primary topic, JET is similar to JST except that each time period has its own model and that STAN based priors are used. Compared with JET, αJET has very low perplexity that does not change much with the increasing number of topics. Such results can be explained because αJET accounts for authors. Therefore, it can be expected that authors do not have many variations in the vocabulary they use since the amount of content posted by each individual is somewhat small.

7.5 Effort Evolution Analysis

The topic and effort labels obtained by applying the αJET model on each of the datasets can be used for providing insights on how effort is distributed within the SF and CO communities. In the following sections, some example of topics and their evolution for the SF and CO community are presented. The evolution of effort across different types of contributors based on their involvement in their communities is also discussed. For each JET model, the following values are set: $K' = 50$ and $K = 10$. For the other parameters, the values defined in section 7.3 are used.

7.5.1 Aggregated Community Effort Evolution

Given the αJET model trained for each community, the evolution of effort within SF and CO is analysed. For such analysis, an approach similar to previous work on community maturity is reused

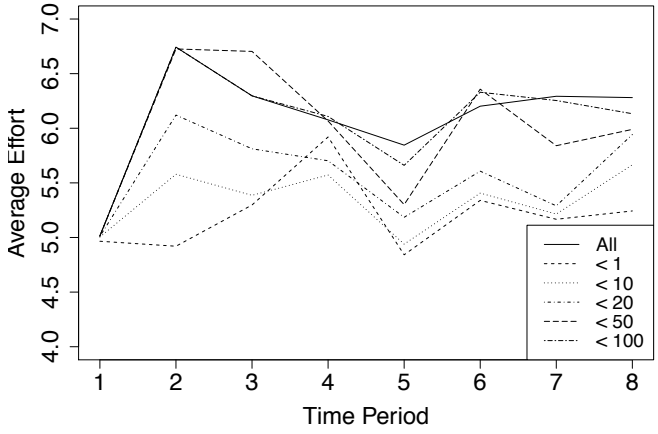


Figure 22: Monthly average contributions effort for different user groups for the *Cooking* (CO) dataset. Lower effort values indicate high effort.

(Chapter 6). In order to better understand the relation between effort and user experience, 5 different groups from content contributors are derived depending on how many days they have been engaged with the community at most. The thresholds are set to (< 1,< 10,< 20,< 50,< 100) which represent that users are engaged with the community for less than one day, less than 10 days, etc. Such community age groups are used as representatives of different user maturity levels. For each community age group $\mathscr{A}' \in \mathscr{A}$ and a time period $p \in \{1 \dots P\}$, the average contribution effort is derived by selecting the subset of documents $\mathscr{D}' \in \mathscr{D}$ that belongs to period p and are authored by one of the contributors $a \in \mathscr{A}'$ using equation 28:

$$E_p(\mathscr{A}') = \frac{\sum_{a \in \mathscr{A}'} \sum_{d \in \mathscr{D}'} \sum_{e=1}^{E} e \times P(e|a,p,d)}{|\mathscr{D}'|} \tag{28}$$

The evolution of the CO community and the SF community are presented in Figure 22 and Figure 23. In both cases, contributors that have been involved in the community tend to have lower effort than the users that have just joined the community. Such observation is particularly visible for SF for the time periods $p \geq 10$. For

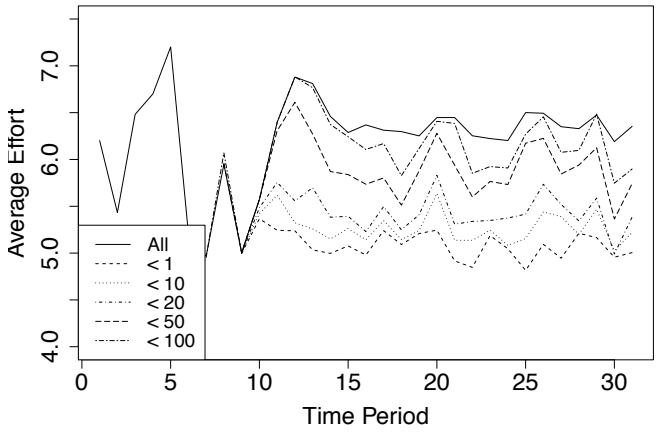


Figure 23: Monthly average contributions effort for different user groups for the *Server Fault* (SF) dataset. Lower effort values indicate high effort.

$p < 10$, there is no clear distinction between the different contributor groups. This is due to the community instability happening during its early months of existence (Section 7.3.1).

Although both communities show that active users have a lower effort compared to the less active ones, each user group have a stable effort within their respective group. As a summary, the more the user community involvement, the less the user effort and users that continue contributing over time are more likely to have a lower effort than others.

7.5.2 *Topic-Effort Evolution Examples*

The focus is now on the evolution of topics in the two communities. Two primary topics $(k'_1, k'_2) \in \{1..K'\}$ are manually selected for each of the communities and the top words of each topic are presented as well as the evolution of the associated labels. Similar to the effort calculation described by equation 28, the mean

effort for words and topics are calculated by aggregating the effort of words associated with each contributors in different periods.

The two topics selected for CO are respectively about *food poisoning* and *cooking temperature* (Table 24). Topic labels are manually set based on the top words of each topic. Table 24 shows that most of the top words in each topic have, on average, relatively low effort. Such result is somewhat expected since the top words of each topic tends to be the most used in the community. As a result, their effort is sensibly lower compared to more scarce terms. Despite a generally low effort some words appears to be more unstable than others over time. For example, *copper* appears in the *food poisoning* topic and tend to be very unstable with low to high effort ($min = 3.412$ and $max = 9$). This may be explained by the subtle relation between *food poisoning* and *copper* (e.g. "*Is hot tap water safe for cooking?*"). An interesting term for the *cooking temperature* topic is *becaus(e)*. Although the term is highly ranked for the topic, the effort tend to be much higher that the other terms. Such value may occur because very few contributors actually know about the reason of using particular food temperature or simply do not take time to explain the reason of a given temperature. Therefore, the term tends to be associated with high effort.

Using a similar approach, the following two topics are extracted for SF: 1) *Domain configuration*, and; 2) *Server performance*. Similarly to the CO topics, relatively stable efforts across the terms of each topic (Table 25) with similar effort values can be observed indicating that both communities have similar word-effort distributions. Contrary to CO, the effort of individual words is much more unstable. Such behaviour is particularly high during the first 10

periods. As showed before, such behaviour may be explained by a community wide contribution instability (Section 7.3.2). Compared with CO, both topics have many terms in common and there is no words with particular high effort in the top words of each topics. Such result may be explained by the fact that many system administrators share the same contribution abilities compared to CO which is more targeted to non professional contributors having greater a disparity in their contribution abilities.

		Average Effort			
Terms	$P(w'\|k')$	Evolution	*Min*	*Max*	*Mean*
chicken	0.02072		5	7.304	6.013
danger	0.01525		4.125	7	5.544
season	0.01056		4.258	7.767	6.219
should	0.009773		4.93	8.006	6.164
bacteria	0.009382		4.2	8	6.021
around	0.008991		4.565	7.407	6.21
potato	0.008991		4.667	8.111	6.249
copper	0.008991		3.412	9	6.109

(a) Topic #3 (Food Poisoning)

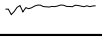
		Average Effort			
Terms	$P(w'\|k')$	Evolution	*Min*	*Max*	*Mean*
temperatur	0.02977		4.274	7.886	6.513
realli	0.01707		4.782	7.674	6.331
becaus	0.01489		3.901	7.454	5.905
should	0.0138		4.93	8.006	6.164
cooker	0.01344		4.5	8.13	6.251
doesn't	0.01271		5.372	7.762	6.113
chicken	0.01198		5	7.304	6.013
thermomet	0.01053		5.064	8.312	6.839

(b) Topic #11 (Cooking Temperature)

Table 24: Top 8 words for two different topics for the *Cooking* (CO) dataset with word average effort evolution. Lower effort values indicate high effort.

As a summary, each community has a similar effort pattern for each topic. However, within topics, some terms seem more costly than others meaning that within topics, only few contributors have the ability to contribute on very specific sub-fields.

Table 25: Top 9 words for two different topics for the *Server Fault* (CO) dataset with word average effort evolution.

		Average Effort			
Terms	$P(w'\|k')$	Evolution	*Min*	*Max*	*Mean*
server	0.03572		4.167	8.332	6.43
address	0.01168		1	8.929	6.374
connect	0.01053		3.714	7.351	6.161
should	0.01036		4.176	7.255	6.194
network	0.009876		3	8.471	6.186
system	0.009754		3.75	8.421	6.362
client	0.009388		1	7.285	6.028
instal	0.009246		4	8	6.361
configur	0.00884		2	7.306	6.151

(a) Topic #4 (Domain Configuration)

		Average Effort			
Terms	$P(w'\|k')$	Evolution	*Min*	*Max*	*Mean*
server	0.03355		4.167	8.332	6.43
network	0.01174		3	8.471	6.186
should	0.01079		4.176	7.255	6.194
connect	0.01065		3.714	7.351	6.161
system	0.009966		3.75	8.421	6.362
perform	0.009537		4	9	6.36
machin	0.008445		3.333	8	6.398
address	0.008328		1	8.929	6.374
memori	0.00786		4.931	7.667	6.187

(b) Topic #7 (Server Performance)

7.6 DISCUSSION

Measuring the amount of work put into individual contributions is a complex task that has attracted very little attention. Nevertheless, the concept of contribution effort is formalised in this chapter and a method that can approximate effort based on word usage patterns is proposed (H1.4).

Due to the lack of available ground truth, it was impossible to directly obtain effort annotation from each post. Although, third party annotators could have been asked to label the effort of different users' contributions, it would have been inappropriate since there is

no accurate way to generate a gold standard manually. Rather than following this approach, this chapter proposed to define three hypotheses that correlate with user effort: 1) Active users contribute with lower effort; 2) Users take more time to respond to questions that require more effort, and; 3) Users' contributions incur lower effort when using familiar vocabulary. Each of them were validated using t-tests. This approach to demonstrate that the proposed models tend to be a good representation of user effort.

By analysing the evolution of effort in two different communities, it appears that actively involved users are more likely to have lower efforts whereas new users require more effort to contribute. This result is in line with what can be expected from veteran contributors that, thanks to their experience, require less time to contribute. Such observation shows that contribution effort can be used for modelling the reactivity of community users (RQ1.4) thus validating the hypothesis that effort is a good measure of community reactivity and that it can be approximated from "the amount of effort required for generating the words that form an answer" (H1.4).

Although these results are hard to compare with previous findings, the results complement the findings obtained in the previous chapter on community *maturity* (Chapter 6). Correlating the result with user *maturity* shows that maturity and expertise are highly related with effort: the more experienced the users, the more mature they are. Since maturity is a proxy measure of content complexity, it can deduced that, in the case of the SF community, difficult questions are preferred by contributors that require less effort to contribute. Similarly, because mature users tend to be experts, it can be concluded that experts have a lower contribution effort.

Two different complementary methods were used for modelling contribution effort. First, a fast statistical approach based on Stanines (STAN/ASTAN) was proposed. Then, STAN/ASTAN were improved by using a generative model that learns topic-level effort of documents. Although both approach were successful at correlating with the proposed hypotheses, only the JET and αJET models are able to learn topic dynamics and deal with newly introduced terms. In this context, STAN/ASTAN can be recommended for time critical tasks whereas JET and αJET can be applied for problems that require more reliable effort estimations and topic-level effort evolution patterns. For instance, due to the computational advantage of ASTAN, ASTAN si used in chapter 8 for predicting *best answers* as the computation of effort of each user is needed for each dataset.

The proposed effort models can be easily applied to different communities and social networks. In particular, it can be used as a proxy for estimating the time and amount of labour that users require for contributing thus helping the identification of valuable content in Q&A communities. Such concept of effort can be useful for different settings such as experts identification and content complexity detection. Although effort cannot replace completely expertise and complexity measures, the models require little information and are particularly suitable for communities that do not support complex metadata.

In this chapter, effort-topic dynamics over time were not modelled explicitly. A possible method would be to use the learned effort-topic-word patterns in previous periods as the priors in effort modelling in the current period, similarly to the DJST [223]. Following an equivalent approach, previous word-level effort bias could be also

[223] *He et al. (2014)*

integrated into the models in conjunction with the STAN/ASTAN dynamic prior. The explicit modelling of effort-topic dynamics is left for future work.

7.7 SUMMARY

In this chapter, the concept of *effort*, a novel approach for measuring the amount of labour users put into their online contributions was introduced. First it was shown that effort can be modelled using Stanines (STAN and ASTAN) by comparing results with a set of hypotheses. Second, in order to overcome the limitation of Stanines in the particular case of newly or recently introduced vocabulary two Bayesian models were introduced. JET and αJET both learn topic-level efforts by incorporating word-level effort prior derived from the proposed Stanine based measures. The proposed models achieve better results compared with the existing LDA and JST models in predictive perplexity while keeping the properties of the Stanine based effort measure. Effort is also related with expertise and maturity and it appears that experienced contributors have lower effort than other users.

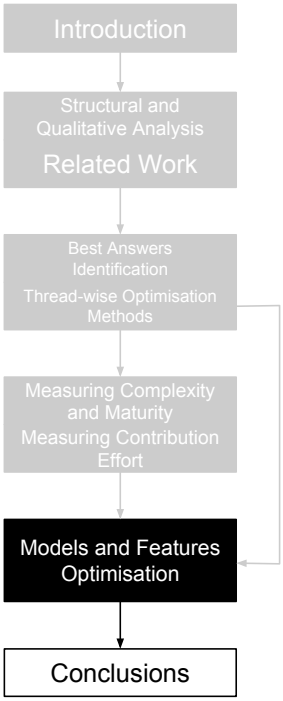
The proposed model of effort showed that by associating effort to the individual words that form a particular contribution, it is possible to model the reactivity of users in Q&A communities (H1.4). Such results shows that contribution effort can effecitvely be used as a proxy measure of user reactivity (RQ1.4).

Although the proposed models were not tested on the SCN community due to the amount of computation required for fully computing JET and αJET, the simpler STAN/ASTAN models can be easily applied. Consequently, ASTAN coupled with the omega metric discussed in the previous chapter are applied in the following chapter (Chapter 8) as the automatic identification of *best answers* is revisited in order to determine if qualitative design improves the identification of *best answers* in Q&A communities (Chapter 9).

8

MODELS AND FEATURES OPTIMISATION

The previous two chapters introduced different measures designed for improving the identification of *best answers* in Q&A communities. This chapter integrates such features in diverse forms into the normalised prediction models discussed in chapter 5 in order to evaluate the best *answer identification* power of features designed using the qualitative design approach studied in the thesis.



Using the complexity metric omega (Chapter 6) and the contribution effort ASTAN measure (Chapter 7), a set of user and content features are derived and new identification models are built. Although the results do not show significant improvements compared the models presented in chapter 5, the normalised features introduced in this chapter appear to be well associated with *best answers* compared to the normalised features previously discussed. These results show that the model presented in chapter 5 is hard to improve upon by simply adding more features and that, rather than improving the identification of *best answers*, the new features

add complexity to the model. Nevertheless, the findings show that features based on user beliefs correlate with *best answers* (H1.2).

This chapter is divided into five sections. First, the motivation to use qualitative design is discussed as well as the contribution of the chapter. Then, the new features derived from the models presented in the last two chapter are presented. In the third section, the new features are added to the normalised model created in chapter 5 and the new model is evaluated. The third section also investigates the importance of the new features by reporting their IGR and, based on these results, feature reduction is performed in order to create more accurate *best answer* identification models. Finally, the results are discussed before summarising them.

8.1 INTRODUCTION

In order to evaluate the ability of features based on qualitative design for improving the identification of *best answers*, the question complexity, maturity metric and different effort models have been developed in the two previous chapters. This chapter focus on the integration of these measures with order ranking normalisation for evaluating if features derived from users beliefs can help *best answer* identification (H1.2) and if such features compare favourably to other *best answer* predictors (RQ1.2).

In this chapter, *user* and *content* features derived from the effort and complexity metrics are used and evaluated by estimating their predictive abilities for identifying *best answers*. Besides adding these

features to the models previously discussed, a feature selection approach is also applied for determining the minimum amount of features and the best predictors required for obtaining better predictions. The goal is to find if only a fraction of features is actually necessary for identifying *best answers* and if the new features and the qualitative design methodology is helpful. Such optimisation is also designed to reduce the amount of computations required for finding *best answers*. Accordingly, the main contributions of this chapter are:

1. Introduce a set of user and content effort and complexity features based on ASTAN and the omega metric.
2. Evaluate the usefulness of such metrics for identifying *best answers*.
3. Investigate the importance of the new metrics compared with the order normalised features discussed in chapter 5 for *best answers* predictions.
4. Perform model optimisation by minimising the number of features required to obtain quality predictions.
5. Investigate if community contributors' belief about what makes quality answers can be used for identifying and designing features (i.e. question complexity, maturity and contribution effort) that help the automatic identification of *best answers* (H1.2).

8.2 Predicting Best Answers with Qualitative Design

Following the feature based approach used in this thesis, and the design of the question complexity, user maturity and contribution effort in the previous two chapters, the qualitative design methodology is evaluated.

In the following subsections, the model used for the evaluation is described as well as the features that are used for building the *best answers* identification model. In particular, all the features used in chapter 5 are reused and a few additional *user* and *content* features are introduced based on the complexity metric introduced in chapter 6 and the effort models discussed in the previous chapter.

8.2.1 Best Answers Models

In chapter 5, different methods for identifying *best answers* were discussed such as non-normalised and normalised classifiers and LTR models. As highlighted in chapter 5, better results were obtained when using such optimisations compared with the non-optimised models described in chapter 4. In this chapter, thread order normalisation is reused as well as the *Alternating Decision Tree* algorithm since they gave good results. Besides reusing such type of model the thread based stratified cross-validation folding evaluation method is applied in order to compare the results presented in this chapter with the results of chapter 5.

	Features Set		
Type	Core Features Set (28)	Extended Features Set (31)	Current Features (24)
User	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio.* (18)
Content	*Answer Age, Number of Question Views, Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (6)	***Number of Comments**, Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (8)	*Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy.* (4)
Thread	*Number of Answers, Answer Position, Relative Answer Position, Topical Reputation Ratio.* (4)	***Score Ratio**, Number of Answers, Answer Position, Relative Answer Position, Topical Reputation Ratio.* (5)	*Answer Position, Topical Reputation Ratio.* (2)

Table 26: List of features and features categories without the complexity and effort features.

8.2.2 *Features Sets*

Similarly to the previous chapter the *users*, *content* and *thread* feature sets are reused. For allowing a better comparison between the new results presented in this chapter and the previous ones, the features described in chapter 5 are reused. For clarity, the different feature sets including the *core*, *extended* and *stable* features sets are reproduced in Table 26.

8.2.3 *Complexity and Maturity Features*

In this chapter, the focus is on evaluating the addition of new features to support the identification of quality answers with a particular focus on complexity and effort features. In this section complexity and maturity features are added. In chapter 6, different models of complexity were discussed and it was found that ML models cannot be defined without having available question complexity annotations. As a consequence, the omega complexity metric (Ω) was introduced. The omega metric can be computed even if no complexity annotations are available and is therefore applicable to most of the datasets studied in this thesis.

User Features In this chapter, user-based complexity and maturity features are considered as they can be used even when only current features are employed since they are computed from previous user observations. As a result, eight different user features based on the omega metric are computed. These features a described as follow:

- *Average Question Complexity:* Measures the average complexity of the questions asked by a given user.
- *Average Answers' Question Complexity:* Measures the average complexity of the questions answered by a given user.
- *Average Posts' Question Complexity:* Measures the average complexity of the questions posted and answered by a given user.

– *Average Solved Question Complexity:* Measures the average complexity associated with the questions asked by a user that have been solved.

– *Average Best Answers' Question Complexity:* Measures the average complexity associated with the questions answered by a user that have been marked as *best answers*.

– *Posting Maturity:* Defines the posting maturity of a user based on the proportion of complex questions answered and posted. This metric follows the definition of maturity presented in chapter 6 and can be calculated using the omega measure (Ω) and considering that complex questions can be identified when $\Omega > 0.5$. At a given time $t \in T$, given a set of asked and answered questions $P_{a,t}$ by a user $a \in A$, the number of complex answered and questions $|P_{a,t}^{\Omega>0.5}|$, the posting maturity $M(P_{a,t})$ can be calculated by:

$$M(P_{a,t}) = \frac{|P_{a,t}^{\Omega>0.5}|}{|P_{a,t}|} \tag{29}$$

– *Asking Maturity:* Define the asking maturity of a user based on the proportion of complex questions asked. It is calculated similarly to the *posting maturity*.

– *Answering Maturity:* Define the answering maturity of a user based on the proportion of complex questions answered. It is calculated similarly to the *posting maturity*.

Content Features The omega metric can be used directly for calculating the complexity of the questions answered by users. We have only one content metric defined as follow:

– *Question Complexity (Ω):* Represent the complexity of questions answered. It is calculated using the omega metric.

8.2.4 Effort Features

Besides the complexity based measures, a set of features based on the effort models presented in chapter 7 are introduced. Since computing αJET is very time consuming, the ASTAN model is used instead. However, rather than using the 1 to 9 Stanine scale, the score is normalised between 0 and 1 using the following formula:

$$ASTAN_{norm}(c) = \frac{ASTAN - 1}{8} \tag{30}$$

User Features Similarly to the user based complexity features, effort measures for users are derived. Five additional user metrics based on the normalised ASTAN measure are defined below:

– *Average Post Effort:* Represent the average normalised effort required for posting content for a given user.

– *Average Answer Effort:* Represent the average normalised effort required for answering questions for a given user.

– *Average Question Effort:* Represent the average normalised effort required for asking questions for a given user.

– *Average Solved Questions Effort:* Measures the average normalised effort associated with the questions asked by a user that have been solved.

– *Average Best Answers Effort:* Measures the average normalised effort associated with the answers posted by a user that have been marked as *best answers*.

	Features Set		
Type	Core Features Set (33)	Extended Features Set (44)	Current Features (39)
User	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio, Post Effort, Answer Effort, Question Effort, Solved Questions Effort, Best Answers Effort.* (23)	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio, Post Effort, Answer Effort, Question Effort, Solved Questions Effort, Best Answers Effort, Question Complexity, Answers Question Complexity, Posts Question Complexity, Solved Question Complexity, Best Answers Question Complexity, Average Post, Posting Maturity, Asking Maturity, Answering Maturity.* (32)	*Reputation, Community Age, Post Rate, Asking Rate, Answering Rate, Normalised Activity Entropy, Number of Posts, Number of Answers, Answers Ratio, Number of Best Answers, Best Answers Ratio, Number of Questions, Questions Ratio, Normalised Topic Entropy, Topical Reputation, Z-score, Question Success, Question Success Ratio, Post Effort, Answer Effort, Question Effort, Solved Questions Effort, Best Answers Effort, Question Complexity, Answers Question Complexity, Posts Question Complexity, Solved Question Complexity, Best Answers Question Complexity, Average Post, Posting Maturity, Asking Maturity, Answering Maturity.* (32)
Content	*Answer Age, Number of Question Views, Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy, Contribution Effort.* (6)	***Number of Comments**, Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy, Contribution Effort, Question Complexity.* (7)	*Number of Words, Gunning Fog Index, Flesch-Kinkaid Grade Level, Term Entropy, Contribution Effort.* (5)
Thread	*Number of Answers, Answer Position, Relative Answer Position, Topical Reputation Ratio.* (4)	***Score Ratio**, Number of Answers, Answer Position, Relative Answer Position, Topical Reputation Ratio.* (5)	*Answer Position, Topical Reputation Ratio.* (2)

Table 27: List of features and features categories including the complexity and effort features (underlined).

Content Features The normalised ASTAN model can be used directly for measuring the effort a user puts into each of her contributions. As with the complexity based metrics, there is only one content metric defined as follow:

– *Contribution Effort (normalised ASTAN):* Represents the contribution effort associated with a particular answer. It is calculated using the normalised ASTAN measure.

8.2.5 New Feature Sets

Compared to the features sets used in chapter 5, 13 new *user* features and two additional *content* features are added. Although *thread* features could be designed, complexity and effort-based thread features are not added as thread order normalisation is applied. As discussed in chapter 5, such a normalisation makes all the features relational. As a consequence, each feature becomes a thread feature and do not require the creation of separated thread features.

Besides not adding new thread features, it is also important to note that complexity based features cannot be computed for the SCN dataset as they require answer rating. Also, content-based complexity features are not stable features as they need information not necessarily available when *best answers* predictions need to be done.

For clarity, Table 27 presents the complete feature set used in this chapter.

8.3 Best Answers Identification using Maturity and Effort

In order to evaluate the importance of the complexity and effort-based features, the experiment with rank order normalisation presented in chapter 5 is performed where the only difference is the addition of new features. In particular, the discussion focuses on the differences between the previous results and the new results in order to highlight the effect of effort and complexity predictors.

Feature reduction is also performed in order to optimise the complexity of the *best answers* identification models by ranking features using IGR and building models incrementally (i.e. feature ablation method). This approach is also used for evaluating the ability of the complexity, maturity and effort features to improve *best answer* identification.

8.3.1 Experimental Setting

For this experiment, the impact of complex and effort-based features on *best answers* predictions is compared for each dataset. The same models used in chapter 5 and the same evaluation approach are performed but the focus is on the analysis of the impact of the predictors introduced in this chapter. Different models are constructed using *Alternating Decision Trees* and thread order rank normalisation is used. The results are also evaluated on different feature subsets and a 10-folds cross validation technique is applied using the thread splitting method (Chapter 5).

As with the previous chapters, the precision (P), recall (R) and the harmonic mean F-measure (F1) are reported as well as the area under the Receiver Operator Curve (ROC) measure for each feature subsets.

Besides this experiment, feature ranking is also performed by computing the IGR of each normalised feature for each dataset and analyse their relative importance for identifying quality answers. Using these rankings, new models are built incrementally by using features subsets obtained from the rankings. Then, it is determined if better results can be obtained by using a restricted amount of features.

8.3.2 Results: Model Comparison

The evaluation results are reported in Table 28. Such results can be compared directly with the Table 15 described in chapter 5. Since the only difference compared with the previous work is the addition of complexity and effort measures, the analysis only focus on them and on the difference between the new and past results. Since the baseline and thread features are not different to the previous experiment, they are not discussed in this chapter.

Core Features: Compared with the experiment performed in Chapter 5, there is additional user and content features. For the core feature sets, there is no complexity features as omega needs answer ratings for being computed. Therefore, the only difference lies in the addition of effort-based metrics.

In general, there is no real difference between the previous results discussed in chapter 5 and the new observations. A negligible increase in F_1 can be observed when using all the normalised features with the effort and complexity features compared to all the normalised features previously studied. Similarly, there is no changes for using most of the core features with a little increase observable for the content features. Although this difference is not significant, the increase in this feature set is due to the presence of effort metric.

The lack of increase in accuracy may be due to the amount of features that are part of the feature sets. Indeed, as observed in the previous chapter, the features used in chapter 4 and 5 already provide accurate predictions. Therefore the addition of complexity, effort and maturity features may not increase the identification of *best answers* even if they correlate with quality answers.

	SCN Forums				Server Fault				Cooking			
Features	*P*	*R*	F_1	*AUC*	*P*	*R*	F_1	*AUC*	*P*	*R*	F_1	*AUC*
Words	0.713	0.713	0.713	0.763	0.727	0.727	0.727	0.771	0.715	0.715	0.715	0.765
Answer Score	-	-	-	-	0.826	0.826	0.826	0.863	0.853	0.853	0.853	0.884
Answer Sc. Ratio	-	-	-	-	0.826	0.826	0.826	0.863	0.853	0.853	0.853	0.884
Users	0.725	0.799	0.760	0.855	0.717	0.767	0.741	0.811	0.675	0.756	0.713	0.793
Content	0.699	0.806	0.749	0.799	0.727	0.765	0.746	0.818	0.691	0.751	0.720	0.790
Threads	0.665	0.847	0.745	0.819	0.721	0.739	0.730	0.804	0.650	0.761	0.701	0.771
All	0.773	0.807	0.790	0.877	0.730	0.777	0.753	0.834	0.723	0.765	0.744	0.826
All-	0.771	0.805	0.788	0.876	0.725	0.781	0.752	0.831	0.719	0.766	0.741	0.824
Users+	-	-	-	-	0.717	0.767	0.741	0.811	0.680	0.745	0.711	0.795
Content+	-	-	-	-	0.825	0.828	0.826	0.901	0.848	0.856	0.852	0.913
Threads+	-	-	-	-	0.826	0.826	0.826	0.903	0.853	0.853	0.853	0.903
All+	-	-	-	-	0.831	0.831	0.831	0.911	0.847	0.856	0.851	0.913
All±	-	-	-	-	0.730	0.777	0.753	0.834	0.723	0.765	0.744	0.826

Table 28: Average answer *Precision*, *Recall*, F_1 and *AUC* for the *SCN Forums*, *Server Fault* and *Cooking* datasets for different feature sets and extended feature sets (marked with +) and reduced features sets (marked with -) using thread order normalisation, complexity-based and effort-based features.

Extended Features: The extended feature sets include the additional complexity metrics. As with the core features set, there is no

clear improvement compared with the previous results. Such result may be explained by the already high accuracy of the models presented in chapter 5.

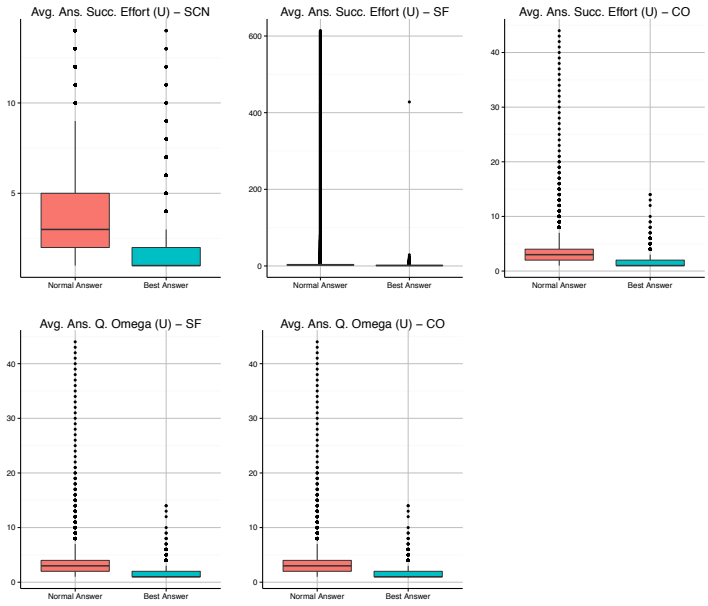


Figure 24: Box Plots representing different order normalised effort and complexity features for the *SCN Forums*, the *Server Fault* and *Cooking* datasets.

Current Features: Similarly to the previous observations, there is no clear advantage of the effort and complexity metrics compared to the usage of the features previously introduced.

8.3.3 *Results: Features Selection*

Following the previous results, the importance of the new features is compared with the predictors studied in the previous chapters. As in the previous experiments, the IGR is calculated and the results are reported in Table 29. For better understanding how such features are ranked across datasets, the rankings of the features are also merged by averaging the order in which each features are ranked (Table 30).

Core Features: The initial focus is on the *core feature* set. As previously discussed, the core features do not contain complexity features since they cannot be computed for the SCN dataset. Although effort features do not appear in the top five features of each datasets, they remain well ranked with 3 effort features featured in the top 20 features for the SCN and SF datasets and 2 features for the CO community. All these features are user features meaning that *best answers* are distinguished from the typical user contribution effort patterns rather than the effort they put into their current contributions.

	SCN		Server Fault		Cooking	
R.	IGR	Feature	IGR	Feature	IGR	Feature
1	0.0763	*Answer Ratio (U)*	0.1532	***Score (C)***	0.1732	***Score (C)***
2	0.0747	*Z-Score (U)*	0.1532	***Score Ratio (T)***	0.1732	***Score Ratio (T)***
3	0.0683	*Reputation (U)*	0.0914	*No. Answers (T)*	0.0793	***No. Comments (C)***
4	0.0681	*No. Answers (U)*	0.0809	***No. Comments (C)***	0.0686	*No. of Words (C)*
5	0.0644	*No. Posts (U)*	0.0765	*Term Entropy (C)*	0.0674	*Term Entropy (C)*
6	0.0607	*No. Bests (U)*	0.0754	*No. of Words (C)*	0.0662	*Avg. Ans. Succ. Effort (U)*
7	0.0588	*No. Answers (T)*	0.0635	*Avg. Ans. Succ. Effort (U)*	0.0651	*No. Bests (U)*
8	0.0550	*Avg. Ans. Succ. Effort (U)*	0.0628	*A. Succ. Ratio (U)*	0.0650	*A. Succ. Ratio (U)*
9	0.0546	*A. Succ. Ratio (U)*	0.0538	*Q. Succ. Ratio (U)*	0.0623	*Reputation (U)*
10	0.0537	*Answering Rate (U)*	0.0527	*Reputation (U)*	0.0619	*No. Answers (T)*
11	0.0536	*Term Entropy (C)*	0.0526	*Avg. Q. Succ. Effort (U)*	0.0532	*Avg. Q. Succ. Effort (U)*
12	0.0527	*No. of Words (C)*	0.0522	*No. Bests (U)*	0.0505	*Answering Rate (U)*
13	0.0510	*Community Age (U)*	0.0500	*No. Posts (U)*	0.0502	*Z-Score (U)*
14	0.0474	*Topic Rep. (U)*	0.0496	*Avg. Ans. Q. Omega (U)*	0.0498	*No. Posts (U)*
15	0.0474	*Topic Rep. Ratio (T)*	0.0496	*Avg. P. Q. Omega (U)*	0.0497	*No. Solved (U)*
16	0.0466	*Post Rate (U)*	0.0495	*Answer Ratio (U)*	0.0485	*No. Answers (U)*
17	0.0464	*Avg. Effort (U)*	0.0482	*Avg. Q. Effort (U)*	0.0483	*No. Questions (U)*
18	0.0443	*Avg. Ans. Effort (U)*	0.0482	*No. Answers (U)*	0.0475	*Avg. Ans. Q. Omega (U)*
19	0.0427	*Position (T)*	0.0477	*No. Solved (U)*	0.0475	*Avg. P. Q. Omega (U)*
20	0.0427	*Rel. Position (T)*	0.0476	*Question Ratio (U)*	0.0474	*Asking Rate (U)*

Table 29: Top normalised features ranked by Information Gain Ratio (IGR) for the *SCN*, *Server Fault* and *Cooking* datasets. Type of feature is indicated by *U*/*C*/*T* for *User*/*Content*/*Thread*.

The *Avg. Ans. Succ. Effort (U)* is the top effort feature for each dataset and is ranked in second position on average just behind the *Nb. of Answers (T)* feature (Table 30). Looking at the distribution of the *Avg. Ans. Succ. Effort (U)* features (Figure 24), it appears that

users that have a low effort are associated with successful answers and are more likely to provide *best answers*. This result confirms some of the previous observations that focused users provide better answers (Chapter 4) as low effort is associated with a more focused vocabulary.

Extended Features: The extended feature sets include the maturity metrics and predictors derived from the question complexity measure. Although no maturity based predictors are listed in the top twenty features (Table 29), there are 3 complexity measures listed in the top 20 features of the SF community and 2 of such features listed in the CO community.

Similarly to the effort features, the user features are more relevant for identifying *best answers*. The most important complexity feature is the *Avg. Ans. Q. Omega (U)*. This feature is listed in 11th position on average (Figure 30). Looking at the distribution of this feature (Figure 24), it can be observed that users that reply on average more complex questions are more likely to provide a *best answer*. This result confirm the hypothesis that more knowledgeable users are more likely to provide better answers.

Although the previous observation show that adding the new features does not improve *best answer* identification, such result does not necessary mean that the new features are not relevant for identifying *best answers*. The IGR analysis shows that some features derived from effort and complexity are well ranked in the top 20

	Core Features		Extended Features	
R.	AR.	Feature	AR.	Feature
1	6.6667	*No. Answers (T)*	1.0000	***Score (C)***
2	7.0000	*Avg. Ans. Succ. Effort (U)*	2.0000	***Score Ratio (T)***
3	7.0000	*Term Entropy (C)*	3.5000	***No. Comments (C)***
4	7.3333	*Reputation (U)*	5.0000	*Term Entropy (C)*
5	7.3333	*No. of Words (C)*	5.0000	*No. of Words (C)*
6	8.3333	*A. Succ. Ratio (U)*	6.5000	*Avg. Ans. Succ. Effort (U)*
7	8.3333	*No. Bests (U)*	6.5000	*No. Answers (T)*
8	10.6667	*No. Posts (U)*	8.0000	*A. Succ. Ratio (U)*
9	12.0000	*Z-Score (U)*	9.5000	*No. Bests (U)*
10	12.6667	*No. Answers (U)*	9.5000	*Reputation (U)*
11	15.6667	*Answer Ratio (U)*	11.0000	*Avg. Q. Succ. Effort (U)*
12	16.3333	*Avg. Q. Succ. Effort (U)*	13.5000	*No. Posts (U)*
13	17.6667	*Answering Rate (U)*	15.5000	*Q. Succ. Ratio (U)*
14	20.0000	*Q. Succ. Ratio (U)*	16.0000	*Avg. Ans. Q. Omega (U)*
15	21.3333	*No. Solved (U)*	17.0000	*No. Solved (U)*
16	21.3333	*Community Age (U)*	17.0000	*No. Answers (U)*
17	23.3333	*Post Rate (U)*	17.0000	*Z-Score (U)*
18	23.3333	*Avg. Q. Effort (U)*	17.0000	*Avg. P. Q. Omega (U)*
19	23.6667	*Topic Rep. (U)*	21.0000	*Asking Rate (U)*
20	24.6667	*Asking Rate (U)*	21.5000	*Answering Rate (U)*

Table 30: Top normalised features ranked by average rank using Information Gain Ratio (IGR) for the *SCN*, *Server Fault* and *Cooking* datasets and the core and extended feature sets Type of feature is indicated by *U*/*C*/*T* for *User*/*Content*/*Thread*.

features and therefore correlate well with *best answers*. Such result shows that qualitative design can be used for designing helpful features for identifying *best answers* (H1.2).

8.3.4 Results: Model Optimisation

Given the average rank of the *best answer* predictors, models are created incrementally by adding features to a basic model based on the ranking observed until all the features are used in order to obtain a simpler and potentially better prediction model.

Such approach helps the identification of the minimum amount of features that are necessary for identifying *best answer* automatically and better understand how *best answers* can be identified in Q&A communities.

In order to identify the best model, the F_1 is reported for each generated model for the core features and extended feature sets.

Core Features Models: As it can be observed in Figure 25, most models need very few features to reach high accuracy. For SCN, the best accuracy is obtained when almost the features are used (31 features) but results that are almost as good are observable when 15 features are used. For SF, *best answers* can be identified with almost no features (with one feature, $F_1 = 0.752$) but the best result is obtained when using 14 features ($F_1 = 0.753$) even though the difference is not significant. For CO, the best result is observable with 31 features ($F_1 = 0.744$) but an almost identical result can be seen when using only 14 features ($F_1 = 0.741$).

On average, it appears that 14 features is the best for identifying *best answers*. Looking at these features, it can be observed that two effort features are selected. This means that effort features are useful for identifying *best answers*.

The particular case of SF shows that high accuracy can be obtained when only selecting the number of answers in a thread. This feature is naturally a good discriminant as answers that are the only answer to a thread are always *best answers* in the training and testing sets. It is important to note that in a real world setting such observation is not necessary true and this particular issue should be investigated in future work.

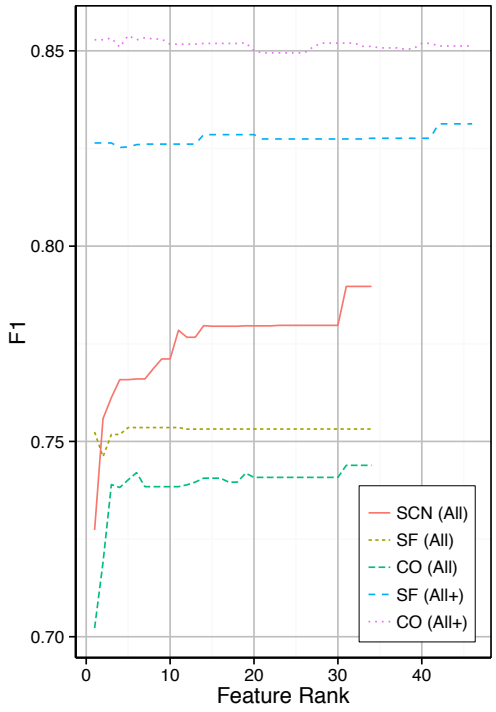


Figure 25: F_1 for the core and extended feature sets for the SCN, SF and CO datasets by incrementing the number of features according to their average IGR ranks.

Extended Features Models: The extended models are dominated by the score feature, therefore, no real improvement can be obtained when adding more features. For SF, the best model is obtained when using 40 features while CO requires 3 features. These results are hard to interpret in relation to the complexity metrics as the predictions models are highly dominated by score features that makes the additional feature not really necessary.

8.4 DISCUSSION

In order to improve the results presented in chapter 4 and chapter 5, different features issued from the qualitative design methodology presented in chapter 2 and complexity, maturity and effort models presented in the last two chapters were investigated in order to identify if features designed from user beliefs can be used

for identifying *best answers* (H1.2) and in order to investigate if these features compare favourably to other *best answers* predictors (RQ1.2).

Although, no significant improvement for identifying *best answers* was obtained when compared to the results presented in chapter 5, some of the newly introduced features correlated well with *best answers* when comparing their IGR showing that such features are to some extent good predictors of *best answers*.

Effort features were particularly well ranked. In particular, it was observed that users that contribute quality answer with low effort are more likely to provide *best answers*. Such result is not surprising as low effort quality answers mean users that are experts and contribute in their preferred domain. Therefore, their new contributions are more likely to be quality answers. This result confirms the user belief that answer reactivity is important for identifying *best answers* (Chapter 2).

Complexity feature were less useful and ranked lower, however, it can be observed that users that answer more complex questions are more likely to provide *best answers*. Again, such result is expected as it was observed in chapter 6 that expert users are more likely to answer complex questions.

Contrary to the previous features, maturity was not ranked very well meaning that the proportion of complex questions answered over time was not as important as expected. Nevertheless, the related average complexity of the question answered was observed as an important feature.

In general, the understanding of the importance of the new features is challenging due to the overwhelming importance of *score* features and the *number of answer in a given thread*. The presence of such features allows for the creation of accurate models without the need of much additional features making the importance of new features hard to understand. This observation prompt the future investigation of simplified models that do not account for scores and other highly ranked features.

8.5 SUMMARY

As part of the qualitative design methodology investigated in this thesis, complexity, maturity and effort metrics were investigated in relation to the identification of *best answers*. Although, improvements in F_1 could not be obtained compared to the results obtained in chapter 5, the ranking of the new features showed that effort and complexity correlate to some extent with *best answers* thus confirming that users beliefs can help the design of *best answers* predictors (1.2). The study of the feature ranks showed that users that create quality answers with low effort are more likely to produce *best answers* and that users that contribute to complex questions are more likely to create quality answers.

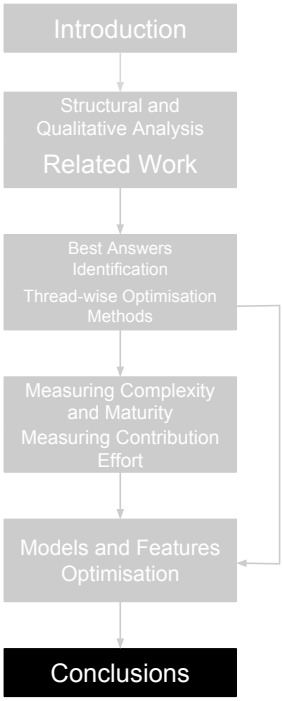
The study of the minimal amount of features required for identifying *best answers* showed the overwhelming importance of *scores* and *the number of answers in answering threads*.

The following chapter reviews the structural and qualitative design methodology presented in this thesis and its applicability to *best answer* identification in Q&A communities. In particular, the strengths and limitation of the approaches are discussed and future work outlined.

Part IV

CONCLUSIONS AND FUTURE WORK

9 CONCLUSIONS



This chapter concludes this thesis by revisiting the different findings, strengths and limitations of the produced research. In particular, the suitability and applicability of the structural and qualitative design approach for improving the identification of *best answers* is discussed in detail as well as the independent usefulness of the newly introduced complexity, maturity and effort models.

The principal goal of this body of work was to determine if structural and qualitative design can help the identification of *best answers* in Q&A communities and if user maturity and contribution effort can be modelled accurately. Two different structural optimisations based on the thread-like structure of Q&A communities were proposed and both proved useful for improving *best answer* identification. As part of the qualitative design methodology, different measures of question complexity, user maturity and contribution effort were investigated. Although, these features did not increase the accuracy of *best answers* identification, it was found that the developed measures effectively correlated with *best answers*.

Besides the previous observations, the different models and features developed as part of this thesis give insights about the applicability of the different obtained results. For example, the results

confirmed the importance of reputation systems and community ratings for identifying best answers. As a consequence it appears that Q&A platform should always include some sort of reputation system. The different model and the new features could also be used effectively by both community manager and contributors for improving the substitutability of their communities by allowing them to find content more effectively or to monitor the health of their community.

During the pursuit of this thesis, it was found that answers *scores* were highly correlated with *best answers*, therefore, future work should investigate if the score of answers can be modelled. Other areas of investigation include the identification of questions without *best answers*, studies of additional communities and the recommendation of questions to answers to contributors.

This chapter is divided in five different sections. First, conclusions concerning the research questions are formulated. Then, in the second section, the strengths and limitations of the research are discussed in details. In the third section, insights and potential industrial applications are discussed. Finally, future areas of work and conclusions are presented.

9.1 Research Questions and Hypotheses Validation

The core research question of this thesis is the application and evaluation of *structural* and *qualitative design* for the automatic identification of *best answers* in online Q&A communities (RQ1). Besides the main contribution, the development of complexity, maturity and effort features is also studied as part of the *qualitative design* methodology. In particular the complexity and maturity models are introduced as a proxy measure of user knowledge (H1.3) while contribution effort models are proposed as a measure of community and user reactivity (H1.4).

In order to ensure that the results can be transferred to different types of Q&A communities, the models, hypotheses and research questions of this thesis are investigated on three different communities that vary in size, structure and topics of interest. For example, the CO communities is a small non-technical community centred on providing culinary advices while SF is a medium sized community focused on computer administration and the SCN forums is a medium sized community of SAP product users supported by Q&A platform with different characteristics (i.e. a forum instead of a dedicated Q&A platform).

The following sections revisit each hypotheses and research questions presented in chapter 1 in order to determine if the postulated hypotheses are valid and if the research questions were investigated thoughtfully. In particular, investigations related to whether

structural and *qualitative* design can improve the performance of automatic identification of *best answers* in Q&A (RQ1).

9.1.1 Structural Design Optimisations

The first approach studied in this thesis was the usage of *structural design* for improving the conception of *best answer* predictors and identification models. The main idea behind the structural design methodology is that the structure of Q&A communities can be used for optimisation purposes.

The main research question related to structural design is whether structural optimisation techniques improve the automatic identification of *best answers* (RQ1.1). Based on some studies of the structural characteristics of Q&A communities in chapter 2. The analysis showed that the studied communities allow for only one *best answer* per answering thread and that Q&A communities are centred around questions that are associated with a set of answers thus forming a thread-like structure. Based on such observations the hypothesis is proposed that structural optimisation techniques that take into account the thread-like structure of Q&A can help the identification of *best answers* (H1.1).

In order to evaluate this hypothesis, two different optimisations were created in chapter 5 and evaluated against the *best answer* identification models introduced in chapter 4.

The first proposed method was to use value relations between the same features of the same thread instead of their raw value. Three

different methods were proposed by building on top of the *thread* features proposed in chapter 4 and the work of Gkotsis et al.[224].

[224] *Gkotsis et al. (2014)*

The second proposed method was to use LTR models for distinguishing the answer that is the most likely the best within the available answers of a thread.

In general, both approaches were successful. Even though there was no dominant winning method between the normalisations approach and the LTR methods. The improvement in identifying *best answers* was significant as on average, for different feature sets, the improvement in F_1 was $+5.3\%$ compared to the models presented in chapter 4.

As a result, the hypothesis that the thread-like structure of Q&A can help the identification of *best answers* was validated (H1.1) and it was shown that structural optimisation techniques improve the automatic identification of *best answers* (RQ1.1).

9.1.2 Qualitative Design Features

The second approach studied was the integration of community members insights into the development of *best answer* predictors. In particular the research question was to investigate how user beliefs about what makes quality answers compare to other *best answer* predictors (RQ1.2). This area of investigation was based on the hypothesis that community contributors' belief can be used for identifying and designing features that help the automatic identification of *best answers* (H1.2).

A user survey conducted as part of this thesis indicated that various features are associated with *best answers* by community contributors (Chapter 2). In particular, two different features were retained and designed. First a measure of question complexity and contributor maturity was designed for modelling user knowledge (Chapter 6). Then, a contribution effort measure was proposed for modelling the reactivity of community users (Chapter 7).

The ability of such models to correlate with *best answers* and to improve the identification of *best answers* was evaluated in chapter 8. Although results did not improve significantly when applying these features designed through qualitative design, it was found that measures like the average effort associated with successfully answered questions was highly correlated with *best answers* thus validating the hypothesis that contributors' belief can be used for identifying and designing features that correlate with *best answers* (H1.2).

Despite not improving on the models presented in chapter 5, some effort and complexity features were ranked well (e.g. *Avg. Ans. Succ. Effort (U)* was ranked 2nd and *Avg. Ans. Q. Omega (U)* was ranked 14th) thanks to a high IGR. Therefore, it can be deduced that the non improvement of the *best answer* identification models are mostly due to the fact that previous features were already good performers. As a result it can be deduced that user beleifs can be used for identifying and designing features that are correlated with *best answers* (H1.2) and that user beliefs about what makes quality answers compare favourably to other *best answer* predictors (RQ1.2).

9.1.3 *Measuring Question Complexity and Maturity*

One of the measures developed as part of the qualitative design methodology was the question complexity model and the derived user maturity model.

The user survey conducted in chapter 2 identified the ability of users to learn new things and their knowledge as an important factor for identifying *best answers*. Rather than simply modelling such a feature directly, this thesis hypothesised that knowledgeable users are more likely to answer or ask complex questions than other users (H1.3) and investigated if question complexity and contributor maturity can be used for measuring the ability of users to learn new things and being knowledgeable (RQ1.3).

In order to model the complexity of questions, questions were annotated as complex or not complex and different models were constructed. Then, based on the findings a complexity metric called omega (Ω) was created and evaluated. Although, the created measures showed a relatively modest precision and recall with an F_1 of 65% on the SF datasets, hypothesis testing showed that high maturity is associated with high reputation. As a consequence, the relation between reputation and user maturity confirmed that knowledgeable users are more likely to answer or ask complex questions than other users (H1.3) and by extension that question complexity and contributor maturity can be used for measuring the ability of users to learn new things and being knowledgeable (RQ1.3).

9.1.4 *Modelling Contribution Effort*

The second model developed following the qualitative design methodology was a set of contribution effort models.

The study performed in chapter 2 observed that the reactivity of community users is important for identifying *best answers*. Instead of only modelling the time-to-answer information which is part of the baseline model created in chapter 4 that only partially accounts for the amount of time a user required for answering questions, different models that account for the implicit amount of work that is put into user contributions were created.

The hypothesis that user reactivity can be estimated from the amount of effort required for generating the words that form an answer (H1.4 was applied in order to create the different effort models presented in chapter 7). This hypothesis was evaluated for understanding if contribution effort can be used for modelling the reactivity of community users (RQ1.4).

In order to model the effort of user contributions, different models were proposed based on the concept of Stanines[225] and the attribution of effort to individual words based on time-to-response information. The evaluation was performed through hypothesis testing and the proposed models were successfully correlated with user reactivity thus confirming that effort can be modelled by taking into account word-effort distributions (H1.4) and that contribution effort can be used for modelling the reactivity of community users (RQ1.4).

[225] *Thorndike (1982)*

9.2 Discussion and Limitations

Although the contributions noted above address the different research questions and hypothesis exposed in chapter 1, these results need further discussion in the general context of *best answer* identification. In the following section the contributions of this body of work are discussed in more details.

9.2.1 General Observations

Although in general the qualitative and structural methods improved the performance of best answer identification models, the results could have been made potentially easier to interpret and more communities could have been analysed if the baseline model developed in Chapter 4 had used less features.

The baseline model presented in Chapter 4 contained 31 different features including 5 thread features. This amount of features generated accurate model that was not improved much by adding new features including the ones developed as part of the qualitative approach followed in this thesis. Although technically the new features appeared relatively useful, the overwhelming importance of *score* features made it harder to understand how particular *best answer* predictors were contributing to a given model. As a consequence a better approach may have been to use much less features as part of the reference model.

Focusing on less features would have also benefited the analysis by reducing the amount of computations required for creating a particular model allowing for more communities to be analysed. For example, in their paper Gkotsis et al.[226] focused on a few features. As a consequence, they were able to scale their model to many communities from the SE network.

[226] Gkotsis et al. (2014)

9.2.2 *Identifying Best Answers with Features Subsets*

The baseline models introduced in chapter 4 includes 32 features divided in *user*, *content* and *thread* features. The model was tested on the three datasets studied in this thesis and the results showed high accuracy with an average F_1 of 0.8173 when all the features are used.

The high accuracy of the result is mostly due to the *score* features and the newly introduced *thread* features. Such features take into account the structure of Q&A communities and are therefore good predictors of *best answers*. The normalisation methods investigated in Chapter 5 are mostly motivated by these encouraging results.

An interesting finding of this work was the lack of correlation between answer length and *best answers* even though some previous work has found that longer answers are correlated with *best answers*.[227] This result may be explained by the difference between the communities studied in previous works and the communities studied in this thesis. The lack of correlation could be due to the variation in the type of questions asked by the studied communities. As questions are different, the length of answers may depend on the context

[227] Jeon et al. (2006); Agichtein et al. (2008)

of each question. As a result the length of *best answers* cannot be used for identifying *best answers* reliably.

The approach presented in chapter 4 does not necessarily contain the best features for identifying *best answers*. In particular some research has shown that n-grams based features are highly relevant for identifying *best answers*.[228] However, such features cannot be interpreted easily meaning that they are highly community dependent and does not give many insights concerning what constitutes *best answers* that can be obtained when using the type of features studied in this thesis. Similarly there are other features that may benefit the identification of *best answers* but were left out as the focus of this thesis is centred on the evaluation of the *structural* and *qualitative design* methodologies.

[228] Agichtein et al. (2008)

As observed, the score features play an important role in the identification of *best answers*. The importance of this feature may be problematic for different reasons: 1) ratings may not always be available when decisions need to be made, and; 2) The importance of answer scores may create over fitted models that do not deal well with cases when *scores* are not relevant. Finally, the models also suffer from a bias on the *number of answers* feature as the models are only tested and trained on questions that have *best answers*. As a consequence, when an answer is the only available answer to a question it will always be identified as *best answer*. Fortunately this issue is partially alleviated as most of the answers of the studied dataset have more than one answer. Nevertheless, future work should investigate the automatic detection of questions that do not have *best answers* in order to avoid this potential bias.

9.2.3 *Thread-wise Optimisation Methods*

As part of the structural design methodology and the observations made in chapter 2, two different structural optimisations where proposed. First different feature normalisation methods where investigated before studying the application of LTR models.

Thread-wise Normalisation Methods: Based on the *thread* features created in chapter 4 and the work by Gkotsis et al.[229] on feature normalisation, chapter 5 proposed distinct features normalisation methods based on the thread-like structure of Q&A communities.

[229] *Gkotsis et al. (2014)*

In general the proposed thread normalisation proved to improve significantly the models presented in chapter 4 by an increase of $+5.3\%$ in F_1 for different feature sets and all the datasets studied in the thesis. Although the approach is not comparable directly to previous work,[230] the proposed approach is different as it generalises the concept of order normalisation to all the predictors presented in chapter 4 and automatically detect which features need to be normalised based on their variance across threads.

[230] *Gkotsis et al. (2014)*

Interestingly it appears that the normalisation approach modifies the importance of features. For example, in chapter 4, the *length of answers* was not correlated with *best answers* but the correlation appears when order normalisation is used. Such a result may be explained by the fact that non normalised features compare the values of predictor across a community whereas thread normalisation localise the comparison of feature values locally (i.e. at the thread level). For example, in the case of *answer length*, long answers at

the thread level are likely to be associated with *best answers* but long answer in general do not distinguish *best answers* from normal answers.

Similarly to the previous observations, other features may benefit more from the normalisation approach. For example, Gkotsis et al.[231] provide some other features such as the longest sentence length that correlate well with *best answers*. Another area of investigation would be to use additional methods for detecting what feature to normalise automatically. One of such approach could be the comparison of IG between normalised and non normalised features.

[231] *Gkotsis et al. (2014)*

Learning To Rank Models: Another approach investigated for optimising *best answer* identification models was the application of LTR models as Q&A communities are organised around threads. The approach used in this thesis was based on a pointwise model based on decision trees. The results were similar to the normalisation approach. Therefore, LTR methods can be applied for improving the identification of *best answers* in Q&A communities.

In this thesis, the LTR models were only applied for identifying the unique *best answer* in a given thread. An important feature of such models is to provide ranked list. In this context it would be interesting to report other traditional LTR metrics such as the MRR to see how far are ranking of the wrongly ranked answers in order to better understand the ability of LTR models to identify *best answers*. Another area of investigation could be the comparison of the results to the more complex pairwise and listwise LTR algorithms to se if better results can be obtained.

9.2.4 *Qualitative Design Features*

For the qualitative design methodology, two distinct area of investigation were selected based on the survey conducted in chapter 2. First, models of question complexity and user maturity were created as a measure of user knowledge (Chapter 6). Then, contribution effort was studied as a proxy measure of community reactivity (Chapter 7). Finally different measures based on these models were derived in chapter 8 in order to evaluate the ability of complexity, maturity and effort metrics to correlate with *best answers*.

Measuring Question Complexity and Maturity The development of a question complexity metric was based on third party annotations of a 510 questions pairs of the SF community. Then, *askers* and *answerers* (users), and; *questions* and *answers* (content) feature were used for training different complexity models before creating a community independent complexity metric called omega (Ω) based on the models findings (Chapter 6). Following the development of the complexity measure, the concept of user maturity was introduced based on the proportion of complex question user ask or answer over time.

Although the accuracy of the models was not very high with an F_1 of 65%, the analysis showed that users that answer complex questions are more likely to have high reputation. Such observation confirmed the ability of user maturity to model user knowledge.

The development of the omega metric (Chapter 6) assumed many different characteristics such as the equal importance each of it constituents. Although the F_1 was similar to the learned model of question complexity, it can be observed that omega as a low precision compared to its ability to recall complex questions. Therefore future work should investigate if the omega metric can be improved for avoiding such pitfall. Future work should also investigate the annotation of additional communities in order to evaluate if other communities really share the same complexity features and if the omega metric performs equal well on other communities.

Modelling Contribution Effort Contribution effort models were created for representing the implicit amount of work users put into their contributions in order to better model the reactivity of contributors (Chapter 7). Since there was no ground truth available and it is impossible to know clearly the amount of work users put into their contributions, the models were evaluated based on hypothesis testing by comparing the models results with expected behaviour. The result showed that the proposed model behave as expected and may be used for representing the reactivity of answerers.

The main advantage of the proposed models is that they can be applied to a large variety of communities as only time-to-response information, authors and textual content is necessary for modelling contribution effort. In this context, it would be interesting to apply such metric to other communities to see how effort varies in different types of online websites.

Perhaps the main issue with some of the proposed models is the high computational cost that is created when using topic models

and authors. This issue is the reason why chapter 8 uses the non topic-model variation of the proposed effort models. Consequently, it appears that the topic model used for predicting contribution effort do not scale very well when multiple and big communities are analysed and that simpler models such as the ASTAN should be preferred when computation time is critical.

Identifying Best Answers with Maturity and Effort Chapter 8 evaluated the correlation between some features derived from the complexity, maturity and effort features and *best answers* in order to determine if features derived from qualitative analysis can help the identification of *best answers*.

In general, the results were not as clear as the structural optimisation methods presented in chapter 5 as the accuracy of the normalised models did not improve significantly when new features were added. Nevertheless, five features were listed in the top fifteen predictors when calculating the IGR and ranking the features by importance. This result highlighted the relative importance of the effort and complexity measures.

The reason for not improving the identification of *best answers* may be explained by the amount of good predictors already used in chapter 5. As a consequence, rather than making the models more accurate, the extra features introduced extra complexity into the *best answer* identification models. As a result, the addition features do not improve the identification process.

In general, the value of the newly introduced features is not limited to the improvement of *best answer* identification in Q&A communities as they can be applied to a wide range of additional datasets. For example, this thesis focused on three different communities. Therefore it would be interesting to apply the features and methods developed in this thesis to more communities. For example, the effort associated with reposting content on Twitter could be estimated using contribution effort. Nevertheless, these methods may not be suitable to any communities. For example effort models would not apply in context where there is no way to measure reaction time (i.e. the elapsed time between an arbitrary event and an related contribution).

9.3 Insights and Applications

Besides the direct applicability of the thesis methodology for improving the design of *best answers* predictors, algorithms and models. The results and features provided in this thesis may be also useful in real-life Q&A platform deployments where they could be used by platform designer, community managers and contributors.

9.3.1 Applications to Community Design

Although it may be useful for Q&A platforms to implement automatic *best answers* identification models, the feature analysis performed in Chapter 8 shows that best answers are mostly correlated with score features. This observation shows that Q&A platforms

should always implement reputation systems and community ratings. The feature ranking provided in Chapter 8 also show that allowing comments on answer is also a good idea.

Concerning the maturity and effort features, effort appeared to be a useful for identifying best answer. As a consequence Q&A platforms could implement such features and present it to potential contributors so they can estimate how much time it will take for them to get answers for a particular question. The maturity measure could be also used for giving some information to contributors about the complexity of individual questions or the ability of particular users (Section 9.3.3).

9.3.2 Applications for Community Managers

As discussed in Chapter 1, the aim of community managers is to ensure the well being of their community so that it thrives. In this context providing methods for helping community manager to determine if their community content is good is important.

The three main contributions of this thesis besides the evaluation of the structural and qualitative design methods are the development of *best answer* identification models and the question complexity, user maturity and contribution effort metrics. Such models and metrics can benefit community managers' work by helping them to better monitor and guide community contributions.

For example, the *best answer* identification models could be used for helping community managers to identify unsolved questions

as well as determine the proportion of questions that are still unanswered. This knowledge could then be used for both labelling *best answers* automatically and soliciting user contributions on particular questions directly.

Question complexity and user maturity may be also useful for helping manager to understand if their community becomes more knowledgeable over time. This particular information is useful as it indicates if a community retains expert users and if contributors become more involved over time. This information could give a better picture of the status of a particular community to its manager.

Finally, the contribution effort metrics may also be used by community managers for identifying the questions that require more time to be answered or simply identify the most relevant answerer for a given question. Then, this information could be used for reaching to particular contributors.

9.3.3 Applications for Community Users

The models and features developed in this thesis may also be useful to community contributor. For example, *best answer* identification models could be used for improving answer retrieval when a user look for a particular answer

Displaying metrics such as the maturity and effort associated with answer to users may also be a good idea as it could give some information to contributors about what questions are more likely to be *quality answers* before any community ratings are provided.

9.4 Future Work

Although different additional work can be investigated in order to extend the work presented in this thesis, a few of possible future areas of investigation are listed in the following sections.

9.4.1 Predicting Community Ratings

As observed in chapter 4 and the other chapters dealing directly with *best answer* identification, *score* features were a good predictor of *best answers*. However, ratings are not necessarily available when deciding to identify *best answers*. In this context the development of rating methods that automatically predict the ratings of answers would be important to investigate.

A possible approach for modelling such issue would be to train regression models that take into account user ability to obtain high ratings and the popularity of a given topic. In order to do so, features introduced in this thesis could be used as well as topic models similar to the one used for contribution effort.

9.4.2 Identifying Non Answered Questions

Another issue observed in chapter 4 is the case of questions that have only one answer. in this case they are always annotated as *best answers*. In order to deal with such an issue a potential future area of investigation could be the identification of questions that do

not have any *best answers*. Using this methodology, questions that do not have *best answer* could be ignored when the *best answer* identification model is applied.

A possible method could be to reuse the *best answer* identification models presented in this thesis and use the likelihood of having *best answers* in answering thread. Then, a model could be trained to correlate the distribution of *best answer* likelihoods with non answered questions.

9.4.3 *Large Scale Best Answer Identification*

In this thesis most of the work was involved on small to medium size communities. A logical continuation of the work would be to investigate similar algorithms on additional communities and large Q&A website. An easy extension would be to perform the same analysis on additional SE communities.

In order to reduce the latency for training the models presented in this thesis, new methods should be investigated. In this thesis some computations were performed in parallel using multi-core processing and some features of the Sparks framework. However, the databases used for storing the data proved to be a bottleneck. A possible optimisation could be the replacement of relational databases to either non relational scalable databases like HBase[232] or Spark SQL[233] which provide an SQL like interface to scalable storage systems.

[232] *HBase*, http://hbase.apache.org.

[233] *Spark SQL*, http://spark.apache.org/sql/.

9.4.4 *Identifying Questions to Answer*

This thesis has mostly investigated the identification of *best answers*. As highlighted in chapter 1, the identification of *best answers* can benefit the identification of questions that have already been answered or the identification of answers that need further contributions.

One of the next step is to identify questions that need replies and find what are the most suitable questions to answer for a particular contributor. However rather than simply recommending questions based on user interests, a better approach is to take into account the answering behaviour of users in order to identify the question they are the most willing to answer. Some initial work has been already carried out towards the identification of question to answers with some success by using LTR models and different feature similar to the one presented in this thesis.[234] However the approach currently requires high computation. Therefore current investigations aim to reduce the complexity of the recommendation model.

[234] Burel et al. (2015a,b)

9.5 Summary and Conclusions

This thesis investigated the definition and evaluation of two distinct methodologies for improving the identification of *best answers* in Q&A communities (RQ1). To this extent this body of work investigated three different Q&A communities that have contrasting characteristics: 1) The CO community is a small Q&A community where contributors share cooking advice; 2) SF is a medium size

community that focus on the administration of computer systems, and; 3) The SCN forums is a medium size supporting community for SAP product users.

In order to improve the identification of *best answers* two different methodologies were proposed. First a structural design approach proposed to use the thread-like structure of Q&A communities for creating two model optimisation methods (H1.1) in order to investigate if structural optimisation techniques can improve *best answer* identification (RQ1.1). Secondly, a qualitative design method proposed to use user surveys in order to guide the development of *best answers* predictors based on contributors beliefs about what constitutes *best answers* (H1.2) in order to observe if features designed from user beliefs are good *best answers* predictors (RQ1.4).

The structural design approach proposed to normalise *best answer* predictors based on the relational order of the values of individual features within threads while the second method proposed to apply LTR models for identifying *best answers* within answering threads. Both methods proved reliable with the order normalisation approach achieving a statistically significant average gain of +5.3% across different feature sets.

The qualitative design methodology identified a few features related to *best answers*. The research presented in this thesis chose to design and evaluate the concept of user ability to answer complex questions by modelling question complexity and user maturity (RQ1.3). Then the concept of user reactivity was studied through the creation of different contribution effort models (RQ1.4). After validating the ability of each model to represent successfully question complexity, user maturity and contribution effort (H1.3 and

H1.4), the models were integrated into a *best answer* identification model. Although, there was no significant improvement in accuracy, the ranking of the additional features showed some important correlation with *best answers* meaning that contribution effort and question complexity are well associated with *best answers*.

In order to evaluate such models, different *user*, *content* and *thread* features sets were investigated and it was found that answer *scores* and the *number of answers* for a given question were good predictor of *best answers*. In the case of the *number of answers* features, the result is due to the way the models were evaluated as all the questions studied in the dataset had *best answer* labels. Therefore, if a question has only one answer, it is necessarily the *best answer*. The scores are designed for identifying *best answers* so it is expected to correlate well with *best answers*. However, such information may not be always available when deciding what answer is the *best answers*. In this context, future work should investigate: 1) The prediction of answer score in order to generate answer ratings when such information is unavailable, and; 2) A method for identifying the questions that do not have any *best answers*.

Besides such future area of investigations, other domains should be investigated in the future such as the study of bigger communities and additional Q&A websites and the automatic identification of questions to answers for particular users.

Finally, it would be interesting to investigate if the structural and qualitative design methodology could be applied more successfully to other types of communities. For example, models for studying the propagation of sentiment on Twitter[235] could integrate the tweet and retweet structure of the community while user surveys could be

[235] *Twitter*, http://twitter.com.

designed for identifying features that can help the measurement of sentiment. Similarly it would be worth studying the application of the complexity, maturity and contribution effort models to domains not related with Q&A communities.

BIBLIOGRAPHY

L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman. Knowledge sharing and yahoo answers: Everyone knows something. In *Proceedings of the 17th International Conference on World Wide Web*, volume 17 of *WWW '08*, pages 665–674, 2008.

E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. Finding high-quality content in social media. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, pages 183–194, 2008.

E. Agichtein, Y. Liu, and J. Bian. Modeling information-seeker satisfaction in community question answering. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 3(2):10, 2009.

M. Anderka, B. Stein, and N. Lipka. Detection of text quality flaws as a one-class classification problem. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 2313–2316. ACM, 2011.

M. Anderka, B. Stein, and N. Lipka. Predicting quality flaws in user-generated content: The case of wikipedia. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 981–990, 2012.

Y. Bachrach, T. Graepel, T. Minka, and J. Guiver. How to grade a test without knowing the answers — a bayesian graphical model for adaptive crowdsourcing and aptitude testing. In I. M. L. S. (IMLS)., editor, *The 29th International Conference on Machine Learning*, volume 29 of *ICML '12*. International Machine Learning Society (IMLS)., International Machine Learning Society (IMLS)., 06 2012.

A. Berger, R. Caruana, D. Cohn, D. Freitag, and V. Mittal. Bridging the lexical chasm: Statistical approaches to answer-finding. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, pages 192–199. ACM, 2000.

J. Bian, Y. Liu, E. Agichtein, and H. Zha. Finding the right facts in the crowd: Factoid question answering over social media. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 467–476, New York, NY, USA, 2008. ACM.

J. Bian, Y. Liu, D. Zhou, E. Agichtein, and H. Zha. Learning to recognize reliable users and content in social media with coupled mutual reinforcement. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 51–60, 2009.

J. Bishop. Enhancing the understanding of genres of web-based communities: the role of the ecological cognition framework. *International Journal of Web Based Communities*, 5(1):4–17, 2009.

D. M. Blei and J. D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*,

ICML '06, pages 113–120, New York, NY, USA, 2006. ACM.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.

M. J. Blooma, A. Y. K. Chua, and D. H.-L. Goh. A predictive framework for retrieving the best answer. In *Proceedings of the 2008 ACM Symposium on Applied Computing*, SAC '08, pages 1107–1111, 2008.

M. J. Blooma, D. Hoe-Lian Goh, and A. Yeow-Kuan Chua. Predictors of high-quality answers. *Online Information Review*, 36 (3):383–400, 2012.

M. Bramer. *Principles of Data Mining*. Springer Science & Business Media, 2013.

P. B. Brandtzæg and J. Heim. User loyalty and online communities: Why members of online communities are not faithful. In *Proceedings of the 2Nd International Conference on INtelligent TEchnologies for Interactive enterTAINment*, INTETAIN '08, pages 11:1–11:10. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007.

J. Brown and M. Eskenazi. Student, text and curriculum modeling for reader-specific document retrieval. In *Proceedings of the IASTED International Conference on Human-Computer Interaction. Phoenix, AZ*, 2005.

G. Burel, P. Mulholland, Y. He, and H. Alani. Modelling question selection behaviour in online communities. In *Proceedings of*

the 24th International Conference on World Wide Web Companion, WWW '15 Companion, pages 357–358. International World Wide Web Conferences Steering Committee, 2015a.

G. Burel, P. Mulholland, Y. He, and H. Alani. Predicting answering behaviour in online question answering communities. In *Proceedings of the 26th Conference on Hypertext and Social Media*, HT '15, 2015b.

J. Burstein and M. Wolska. Toward evaluation of writing style: Finding overly repetitive word use in student essays. In *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 1*, EACL '03, pages 35–42, 2003.

J. Burstein, K. Kukich, S. Wolff, C. Lu, M. Chodorow, L. Braden-Harder, and M. D. Harris. Automated scoring using a hybrid feature identification technique. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98, pages 206–210, Stroudsburg, PA, USA, 1998. Association for Computational Linguistics.

B. S. Butler. Membership size, communication activity, and sustainability: A resource-based model of online social structures. *Information Systems Research*, 12(4):346–362, 2001.

C. S. Campbell, P. P. Maglio, A. Cozzi, and B. Dom. Expertise identification using email communications. In *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, CIKM '03, pages 528–531, New York, NY, USA, 2003. ACM.

K. Chai, V. Potdar, and T. Dillon. Content quality assessment related frameworks for social media. In *Computational Science and Its Applications–ICCSA 2009*, pages 791–805. Springer, 2009.

J. Chang and D. M. Blei. Relational topic models for document networks. In *International Conference on Artificial Intelligence and Statistics*, pages 81–88, 2009.

B.-C. Chen, J. Guo, B. Tseng, and J. Yang. User reputation in a comment rating environment. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 159–167. ACM, 2011.

M. Chodorow and C. Leacock. An unsupervised method for detecting grammatical errors. In *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference*, NAACL 2000, pages 140–147, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

A. Y. Chua and S. Banerjee. So fast so good: An analysis of answer quality and answer speed in community question-answering sites. *Journal of the American Society for Information Science and Technology*, 64(10):2058–2068, 2013.

C. Cortes, C. Cortes, and V. Vapnik. Support-vector networks. *MACHINE LEARNING*, 20:273–297, 1995. doi: 10.1.1.15.9362.

B. Dom and D. Paranjpe. A bayesian technique for estimating the credibility of question answerers. In *SDM*, pages 399–409. SIAM, 2008.

G. Dror, Y. Koren, Y. Maarek, and I. Szpektor. I want to answer; who has a question?: Yahoo! answers recommender system. In

Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 1109–1117. ACM, 2011.

F. Farmer and B. Glass. *Building web reputation systems*. Yahoo Press, 2010.

P. Fichman. A comparative assessment of answer quality on four question answering sites. *Journal of Information Science*, 37(5): 476–486, 2011.

R. Flesch. A new readability yardstick. *Journal of applied psychology*, 32(3):221–233., June 1948.

G. Forman. An extensive empirical study of feature selection metrics for text classification. *J. Mach. Learn. Res.*, 3:1289–1305, Mar. 2003. ISSN 1532-4435.

Y. Freund and Y. Freund. The alternating decision tree learning algorithm. *IN MACHINE LEARNING: PROCEEDINGS OF THE SIXTEENTH INTERNATIONAL CONFERENCE*, pages 124–133, 1999. doi: 10.1.1.116.2945.

M. Frické and D. Fallis. Indicators of accuracy for answers to ready reference questions on the internet. *Journal of the American Society for Information Science and Technology*, 55(3):238–245, 2004.

R. Gazan. Specialists and synthesists in a question answering community. *Proceedings of the American Society for Information Science and Technology*, 43(1):1–10, 2006.

G. Gkotsis, K. Stepanyan, C. Pedrinaci, J. Domingue, and M. Liakata. It's all in the content: State of the art best answer

prediction based on discretisation of shallow linguistic features. In *Proceedings of the 2014 ACM Conference on Web Science*, WebSci '14, pages 202–210. ACM, 2014.

R. Gunning. *The Technique of Clear Writing*. McGraw-Hill Book Co., New York, 1952.

K. Gwet. Handbook of inter-rater reliability: How to estimate the level of agreement between two or multiple raters. *Gaithersburg, MD: STATAXIS Publishing Company*, 2001.

M. A. Hall. Correlation-based feature selection for discrete and numeric class machine learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pages 359–366, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. ISBN 1-55860-707-2.

F. M. Harper, D. Moy, and J. A. Konstan. Facts or friends?: Distinguishing informational and conversational questions in social q&a sites. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 759–768. ACM, 2009.

F. M. Harper, J. Weinberg, J. Logie, and J. A. Konstan. Question types in social q&a sites. *First Monday*, 15(7), 2010.

D. Hasan Dalip, M. André Gonçalves, M. Cristo, and P. Calado. Automatic quality assessment of content created collaboratively by web communities: A case study of wikipedia. In *Proceedings of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '09, pages 295–304, New York, NY, USA, 2009. ACM.

Y. He, C. Lin, W. Gao, and K.-F. Wong. Dynamic joint sentiment-topic model. *ACM Trans. Intell. Syst. Technol.*, 5(1):6:1–6:21, Jan. 2014.

L. Hirschman and R. Gaizauskas. Natural language question answering: the view from here. *Natural Language Engineering*, 7 (04):275–300, 2001.

J. Jeon, W. B. Croft, J. H. Lee, and S. Park. A framework to predict the quality of answers with non-textual features. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 228–235. ACM, 2006.

B. John, A. Chua, and D. H.-L. Goh. What makes a high-quality user-generated answer? *Internet Computing, IEEE*, 15(1):66–71, Jan 2011. ISSN 1089-7801. doi: 10.1109/MIC.2011.23.

P. Jurczyk and E. Agichtein. Discovering authorities in question answer communities by using link analysis. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 919–922. ACM, 2007a.

P. Jurczyk and E. Agichtein. Hits on question answer portals: Exploration of link analysis for author ranking. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 845–846. ACM, 2007b.

A. Kaplan and M. Haenlein. Users of the world, unite! the challenges and opportunities of social media. *Business horizons*, 53 (1):59–68, 2010.

P. Katerattanakul and K. Siau. Measuring information quality of web sites: development of an instrument. In *Proceedings of the 20th international conference on Information Systems*, pages 279–285. Association for Information Systems, 1999.

J. Kietzmann, K. Hermkens, I. McCarthy, and B. Silvestre. Social media? get serious! understanding the functional building blocks of social media. *Business Horizons*, 2011.

S. Kim and S. Oh. Users' relevance criteria for evaluating answers in a social q&a site. *Journal of the American Society for Information Science and Technology*, 60(4):716–727, 2009.

S. Kim, J. S. Oh, and S. Oh. Best-answer selection criteria in a social q&a site from the user-oriented relevance perspective. *Proceedings of the American Society for Information Science and Technology*, 44(1):1–15, 2007.

J. Kincaid, R. Fishburn, R. Rogers, and B. Chissom. Derivation of new readability formulas for navy enlisted personnel (research branch report 8-75). *Memphis, TN: Naval Air Station, Millington, Tennessee*, page 40, 1975.

V. Kitzie and C. Shah. Faster, better, or both? looking at both sides of online question-answering coin. *Proceedings of the American Society for Information Science and Technology*, 48(1):1–4, 2011.

J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, Sept. 1999a.

J. M. Kleinberg. Hubs, authorities, and communities. *ACM Comput. Surv.*, 31(4es), Dec. 1999b.

S.-a. Knight and J. M. Burn. Developing a framework for assessing information quality on the world wide web. *Informing Science: International Journal of an Emerging Transdiscipline*, 8(5):159–172, 2005.

Y. W. Lee, D. M. Strong, B. K. Kahn, and R. Y. Wang. Aimq: a methodology for information quality assessment. *Information & management*, 40(2):133–146, 2002.

B. Li, T. Jin, M. R. Lyu, I. King, and B. Mak. Analyzing and predicting question quality in community question answering services. In *Proceedings of the 21st International Conference Companion on World Wide Web*, WWW '12 Companion, pages 775–782, 2012.

C. Lin and Y. He. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 375–384. ACM, 2009.

J. Lin and D. Demner-Fushman. Methods for automatically evaluating answers to complex questions. *Inf. Retr.*, 9(5):565–587, Nov. 2006.

J. Lin and P. Zhang. Deconstructing nuggets: The stability and reliability of complex question answering evaluation. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 327–334. ACM, 2007.

C. X. Ling, J. Huang, and H. Zhang. Auc: A statistically consistent and more discriminating measure than accuracy. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, IJCAI'03, pages 519–524, San Francisco, CA, USA, 2003. Morgan Kaufmann Publishers Inc.

T.-Y. Liu. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 3(3):225–331, Mar. 2009.

Y. Liu and E. Agichtein. You've got answers: towards personalized models for predicting success in community question answering. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 97–100. Association for Computational Linguistics, 2008.

Y. Liu, J. Bian, and E. Agichtein. Predicting information seeker satisfaction in community question answering. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 483–490. ACM, 2008.

Y. Liu, A. Niculescu-Mizil, and W. Gryc. Topic-link lda: Joint models of topic and author community. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 665–672. ACM, 2009.

L. Mamykina, B. Manoim, M. Mittal, G. Hripcsak, and B. Hartmann. Design lessons from the fastest q&a site in the west. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 2857–2866. ACM, 2011.

W. Mangold and D. Faulds. Social media: The new hybrid element of the promotion mix. *Business horizons*, 52(4):357–365, 2009.

M. Mathioudakis, N. Koudas, and P. Marbach. Early online identification of attention gathering items in social media. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, pages 301–310. ACM, 2010.

G. H. McLaughlin. Smog grading: A new readability formula. *Journal of reading*, 12(8):639–646, 1969.

K. K. Nam, M. S. Ackerman, and L. A. Adamic. Questions in, knowledge in?: A study of naver's question answering community. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 779–788, 2009.

L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999.

A. Pal and J. A. Konstan. Expert identification in community question answering: exploring question selection bias. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1505–1508. ACM, 2010.

A. Pal, F. M. Harper, and J. A. Konstan. Exploring question selection bias to identify experts and potential experts in community question answering. *ACM Trans. Inf. Syst.*, 30(2):10:1–10:28, May 2012.

U. Paquet. *Bayesian inference for latent variable models*. PhD thesis, Citeseer, 2007.

C. E. Porter. A typology of virtual communities: A multi-disciplinary foundation for future research. *Journal of Computer-Mediated Communication*, 10(1):00–00, 2004.

J. Preece. Sociability and usability in online communities: Determining and measuring success. *Behaviour & Information Technology*, 20(5):347–356, 2001.

J. Quinlan. C4. 5: Programs for empirical learning, 1993.

D. Raban and F. Harper. Motivations for answering questions online. *New media and innovative technologies*, 73, 2008.

S. E. Robertson and S. Walker. On relevance weights with little relevance information. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '97, pages 16–24. ACM, 1997.

M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, UAI '04, pages 487–494. AUAI Press, 2004.

M. Rowe and H. Alani. What makes communities tick? community health analysis using role compositions. In *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*, pages 267–276. IEEE, 2012.

M. Rowe and H. Alani. Mining and comparing engagement dynamics across multiple social media platforms. In *Proceedings of the 2014 ACM Conference on Web Science*, WebSci '14, pages 229–238, 2014. ISBN 978-1-4503-2622-3. doi: 10.1145/2615569.2615677.

M. Rowe, H. Alani, S. Angeletou, and G. Burel. Report on social, technical and corporate needs in online communities. Technical Report 3.1, ROBUST, 2011a.

M. Rowe, S. Angeletou, and H. Alani. Anticipating discussion activity on community forums. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 315–322, Oct 2011b. doi: 10.1109/PASSAT/SocialCom.2011.215.

C. V. Ruiz, L. M. Aiello, and A. Jaimes. Modeling dynamics of attention in social media with user efficiency. *EPJ Data Science*, 3(1):5, 2014.

P. Serdyukov, H. Rode, and D. Hiemstra. Modeling multi-step relevance propagation for expert finding. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1133–1142. ACM, 2008.

C. Shah and J. Pomerantz. Evaluating and predicting answer quality in community qa. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 411–418. ACM, 2010.

J. Sterne. *Social media metrics: How to measure and optimize your marketing investment*. John Wiley & Sons, 2010.

M. Steyvers and T. Griffiths. Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440.

K. Sun, Y. Cao, X. Song, Y.-I. Song, X. Wang, and C.-Y. Lin. Learning to recommend questions based on user ratings. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 751–758. ACM, 2009.

M. Surdeanu, M. Ciaramita, and H. Zaragoza. Learning to rank answers on large online qa collections. 2008.

M. A. Suryanto, E. P. Lim, A. Sun, and R. H. Chiang. Quality-aware collaborative question answering: methods and evaluation. In *Proceedings of the second ACM international conference on web search and data mining*, pages 142–151. ACM, 2009.

R. L. Thorndike. Applied psychometrics. *Behavioural and Cognitive Psychotherapy*, 12(3):390–399, July 1982.

Q. Tian, P. Zhang, and B. Li. Towards predicting the best answers in community-based question-answering services. In *ICWSM*, 2013.

S. L. Toral, M. Rocío Martínez-Torres, F. Barrero, and F. Cortés. An empirical study of the driving forces behind online communities. *Internet Research*, 19(4):378–392, 2009.

R. Vatrapu, D. Suthers, and R. Medina. Usability, sociability, and learnability: A CSCL design evaluation framework. In *Proceedings of the 16th international conference on computers in education (ICCE 2008)*, 2008.

C. Wagner, M. Rowe, M. Strohmaier, and H. Alani. Ignorance isn't bliss: An empirical analysis of attention patterns in online communities. In *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*, pages 101–110, Sept 2012a.

C. Wagner, M. Rowe, M. Strohmaier, and H. Alani. What catches your attention? an empirical study of attention patterns in community forums. 2012b.

P. Welinder, S. Branson, P. Perona, and S. J. Belongie. The multidimensional wisdom of crowds. In *Advances in neural information processing systems*, pages 2424–2432, 2010.

M. Wu. The community health index. *In Proceedings of the 4th International Conference on Persuasive Technology*, 2009.

L. Yang, S. Bao, Q. Lin, X. Wu, D. Han, Z. Su, and Y. Yu. Analyzing and predicting not-answered questions in community-based question answering services. In *AAAI*, 2011.

J. Zhang, M. S. Ackerman, and L. Adamic. Expertise networks in online communities: Structure and algorithms. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 221–230, 2007.

Z. Zhu, D. Bernhard, and I. Gurevych. *A multi-dimensional model for assessing the quality of answers in social Q&A sites*. PhD thesis, 2009.